

TARTU ÜLIKOOL  
Loodus- ja täppisteaduste valdkond  
Füüsika Instituut

Aksel Allas

**SATELLIIDIPILTIDELT PÕLLUKULTUURIDE PIKSELHAAVAL  
KLASSIFITSEERIMINE 3D SIDUMNÄRVIVÕRGUGA**

Bakalaureusetöö (6 EAP)

Füüsika

Juhendajad:  
PhD Kaupo Voormansik  
MSc Indrek Sünter

Tartu 2019

## **Satelliidipiltidelt põllukultuuride pikselhaaval klassifitseerimine 3D sidumnärvivõrguga**

3D sidumnärvivõrgu rakendamine põllukultuuride pikselhaaval klassifitseerimiseks on paljulubav. Kasutades kõigest viit Sentinel-2 pilti maikuust, oli võimalik saavutada 82,92%-line klassifitseerimise kogutäpsus 28 klassi jaoks. Lisaks suutis treenitud närvivõrk väga hästi tuvastada põldude piire. Edasine töö sel teemal peaks kaasama rohkem sisendeid nagu Sentinel-1 radarsatelliidipildid ja meteoandmed.

Tulevikku vaatavalt, defineeriti käesolevas töös optimaalne uurimisala Sentinel-1 ja Sentinel-2 satelliidipiltide kombineerimiseks ning valiti komplekt Sentinel-2 vegetatsiooniindekseid, mis suurendades närvivõrgu sisendtunnuste hulka suurendavad klassifitseerimistäpsust. Lisaks veenduti parimas närvivõrgu treenimise meetodis, milleks osutus treenimine 500 sisendi peal õpisammude vahel.

Märksõnad: Kaugseire, tehisintellekt, süvaõpe, 3D CNN, sidumnärvivõrk, satelliit, Sentinel-2, põllukultuurid, klassifitseerimine

CERCS: T181 — Kaugseire

## **3D CNN for pixel-wise classification of crops using satellite imagery**

The application of 3D CNN for pixel-wise crop classification is very promising. By using just 5 Sentinel-2 satellite images from the month of May, an overall accuracy of 82,92% was achieved for 28 classes. In addition, the trained neural network managed to predict the boundaries of fields very well. Next tasks in this could incorporate more inputs like Sentinel-1 radarr satellite imagery and meteo data.

In a broader perspective, an area of interest was defined for optimally combining Sentinel-1 and Sentinel-2 satellite imagery. Furthermore, a set of vegetation indices were chosen for improving classification accuracy. Lastly, two strategies for training the 3D CNN network were tested and the method of doing a step of training every 500 samples was discovered to be more effective.

Märksõnad: Remote sensing, artificial intelligence, deep learning, 3D CNN, satellite, Sentinel-2, agricultural crops, classification

CERCS: T181 — Remote sensing

# Sisukord

<b>Sissejuhatus</b>	<b>5</b>
<b>1 Teoreetilised alused</b>	<b>7</b>
1.1 Sentinel satelliidid . . . . .	7
1.2 Optiline kaugseire . . . . .	8
1.2.1 Valguse vastasmõju ainega . . . . .	9
1.2.2 Kaugseire mõõtetulemusi iseloomustavad suurused . . . . .	10
1.2.3 Optilises kaugseires mõõdetavad lainepikkuste vahemikud . . . . .	11
1.2.4 Vegetatsiooniindeksid . . . . .	13
1.3 Masinõpe . . . . .	14
1.4 Tehisnärvivõrgud . . . . .	14
1.4.1 Logistiline regressioon . . . . .	15
1.4.2 Aktivatsioonifunktsioon . . . . .	15
1.4.3 Süvaõppe närvivõrgud . . . . .	16
1.4.4 Pärilevi . . . . .	16
1.4.5 Kaofunktsioon . . . . .	17
1.4.6 Kulufunktsioon . . . . .	18
1.4.7 Tagasilevi . . . . .	18
1.4.8 Gradientlaskumine . . . . .	19
1.5 Sidumnärvivõrk ehk CNN . . . . .	19
1.5.1 Sidumoperatsioon ja sidumkiht . . . . .	20
1.6 Tensorflow ja Keras . . . . .	21
<b>2 Kaasaegsed masinõppe arhitektuurid</b>	<b>22</b>
2.1 Ülevaateartiklid . . . . .	22
2.2 Masinõppe arhitektuuri valimine . . . . .	23
<b>3 Lähteandmed</b>	<b>26</b>
3.1 Optimaalse uurimisala defineerimine . . . . .	26

3.2	Eristatavad põllukultuurid . . . . .	28
3.3	Projekti piiritlemine . . . . .	29
3.3.1	Andmete normeerimine, peegelduste probleem . . . . .	29
3.4	Uute indeksite otsimine . . . . .	30
<b>4</b>	<b>Implementatsioon</b>	<b>33</b>
4.1	Töökeskkonna leidmine . . . . .	33
4.2	Keskkonna ülesseadmine . . . . .	33
4.3	Lõplik tulemus GitHubis . . . . .	34
4.4	Andmete eeltöötlus . . . . .	34
4.5	3D sidumnärvivõrgu näitest lähtumine . . . . .	37
<b>5</b>	<b>Tulemused</b>	<b>39</b>
5.1	Põllukultuuride liigitamine 3D sidumnärvivõrguga . . . . .	39
5.1.1	Klassifitseerimise kogutäpsuse baastase . . . . .	39
5.1.2	Esialgne närvivõrgu treenimine . . . . .	39
5.1.3	Närvivõrgu treenimine tehes ühe sammu kogu andmekogu nägemise peale	41
5.1.4	Närvivõrgu treenimine tehes ühe sammu 500 piksli nägemise peale . .	42
5.2	Arutelu . . . . .	46
	<b>Kokkuvõte</b>	<b>47</b>
	<b>Tänuavaldused</b>	<b>48</b>
	<b>Kirjandus</b>	<b>49</b>
<b>A</b>	<b>Lisad</b>	<b>56</b>
A.1	Tehisavaradari tööpõhimõte . . . . .	56
A.2	Keskkonna ülesseadmine . . . . .	58
A.3	Klassifitseeritavad klassid . . . . .	59
A.4	Github Kaustade struktuur . . . . .	60
A.5	Töös kasutatud normeerimine . . . . .	61
A.6	Tõeväärtuste eeltöötlus SNAPi gpt käsureaga. . . . .	62
A.7	Normeerimata veamaatriks . . . . .	63
A.8	Lõpliku mudeli ennustused . . . . .	63
	<b>Lihtlitsents</b>	<b>67</b>



# Sissejuhatus

Ülemaailmse rahvastiku kasvu tõttu on globaalse põllumajanduse iga-aastane saak viimase 50 aastaga kahekordistunud [1] ja ennustatakse, et aastaks 2050 on vaja taas saagi kahekordistumist [2],[3]. Senise saagi kasvust ligikaudu 12 % on tulenenud põllumaa alade laiendamisest ning ülejäänud kasv on saavutatud masinate, uute suurte saagikustega põllukultuuride, keemiliste väetiste, pestitsiidide ja niisutussüsteemide kasutusele võtmisest [1],[4].

Kuigi tänapäevane põllumajandus on edukalt suurendanud toiduainete tootmist, on kaasnenum ka laialdane ökoloogiline kahju. Väetise lai kasutuselevõtt on viinud vee kvaliteedi languseni paljudes piirkondades [5] ning mitmed liigid on levialade hävimise tõttu väljasuremise ohus [6]. Niisutamisega muutuvad ümbruskonna mullad järk-järgult soolaseks, mistõttu iga-aastaselt on ülemaailmselt ligikaudu 1,5 miljon hektarit põllumaad muutumas kasutuskõlbmatuks [7]. Põllumajanduse saagikusele mõjub negatiivselt ka tolmeldajate, eriti mesilaste, elupaikade vähenemine [8]. Lühidalt, on oht, et tänapäevases põllumajanduses saavutatakse lühiajalise saagikuse suurendamine, jättes arvestamata pikaajalised ökoloogilised kaotused [1].

Kasvava vajaduse rahuldamiseks on tarvis kasutada Maa loodusressursse tõhusamalt, leides tasakaalu niisutussüsteemide, keemiliste väetiste ja pestitsiidide kasuliku mõju ja nendega kaasnevate keskkonna muutuste vahel. Soovitud efekti saavutamiseks on tarvis riiklikul ning rahvusvahelisel tasemel rohkem täpset informatsiooni, millega saaks otsustusprotsessi parendada. [1] Kiirelt ning odavalt kättesaadav põllukultuuride kaart, koos mulla- ja meteoandmetega võimaldaks edukalt ennustada ja planeerida nii lühi- kui pikaajalist põllumajanduse kasvu.

Arengustrateegiate parendamiseks vajaliku informatsiooni kogumiseks kasutatakse kaugseire tehnoloogiaid nagu tehisavaradarid, optilised sateliidid, LIDAR-id. Kaugseire võimaldab piisava ajalise ja ruumilise lahutusega vaatlusi, et identifitseerida, kaardistada ja monitoorida maa-alasid ja ilmastikuolusid, võimaldades täpset ja kiiret otsustuste langetamist [9]. Tulenevalt kaugseire andmete hulgast on neid otstarbekas masinõppe meetoditega käsitleda. Masinõpet on edukalt kasutatud näiteks vesiviljeluse [10], toiduainete kindlustamise [11],[12] klimatoloogia [13] ja majandusteaduste [14],[15],[16] uurimiste rakendustes.

Viimase kümne aastaga on masinõppe kiiresti arenenud ja masinõppes on eriti edukaks osutunud

süvanärvivõrgud. Näiteks sidumnärvivõrgud (ingl. k. Convolutional Neural Networks - CNN-id) on osutunud väga sobivaks arvutinägemise rakendustes [17],[18], objektide tuvastamisel [19],[20] ja semantilisel segmenteerimisel, ehk pildi osade jagamisel klassidesse [21],[22]. Viimase aja suurenenud andmete kättesaadavus ning arvutusvõimsuse kasv ajendab masinõpet kasutama ka kaugseires ja põllumajanduses [23], [24].

Käesoleva töö eesmärk on uurida põllukultuuride pikselhaaval klassifitseerimise meetodit ja saavutatavaid täpsusi Eesti tingimustes 3D sidumnärvivõrguga. Uurimisalaks on Lääne-Virumaa ja lähteandmeteks Sentinel satelliitide pildid.

Euroopa põllumajanduse majandusliku ja ökoloogilise jätkusuutlikkuse arendamiseks on võimalik Eesti põlluharijatel taotleda toetusi Põllumajanduse Registre ja Informatsiooni Ametist (edaspidi PRIA). [25]. Põlluharijad on kohustatud toetuste saamiseks täitma nõudeid, milleks on vastavalt toetuse iseloomule näiteks:

- Põllukultuuride mitmekesistamine, ehk mitme kultuuri kasvatamine talupidamises.
- Rohumaade tähtajaline niitmine.
- Põllumassiivides ökoloogiliselt kasulike rohealade säilitamine.

Toetuste nõuetele vastavuse kontrollimiseks teeb PRIA pistelisi kohapealseid kontrole. Käesolev töö on osa suuremast arendustööst, mis lõppfaasis muudab põldude kontrollimise tõhusamaks - põllukaardi abil saab valida pisteliseks kohapealseks kontrolliks põllud, mis kõige tõenäolisemalt rikuvad toetuste nõudeid.

Uurimus on tehtud koostöös PRIA ning OÜ KappaZetaga. Töös proovitakse vastata uurimisküsimustele:

- Kuidas tasub valida Eestis huviala Sentinel satelliitidega põllukultuuride klassifitseerimise uurimiseks, et luua kõige suurem andmestik vähimate satelliidipiltide arvuga?
- Kuidas on mõttekas kombineerida satelliidipilte, et luua parem sisendandmete tunnuskomplekt?
- Millist klassifitseerimistäpsust on võimalik saavutada 3D sidumnärvivõrgu arhitektuuriga ning milline närvivõrgu treenimise strateegia on optimaalne?

Käesolevas töös esmalt käsitletakse optilise kaugseire ning masinõppe teoreetilisi aluseid. Seejärel antakse ülevaade kaasaegsetest masinõppe tulemustest ning ülevaate alusel põhjendatakse 3D sidumnärvivõrgu valikut. Kolmandas peatükis defineeritakse andmestik, tutvustatakse seda ning valitakse komplekt täiendavaid indekseid närvivõrgu sisendiks. Järgmises peatükis antakse ülevaade töökeskkonna ülesseadmisest ning andmete eeltöötlustest. Lõpuks võrreldakse kahte närvivõrgu treenimise strateegiat, lahatakse tulemusi ning tehakse kokkuvõte.

# Peatükk 1

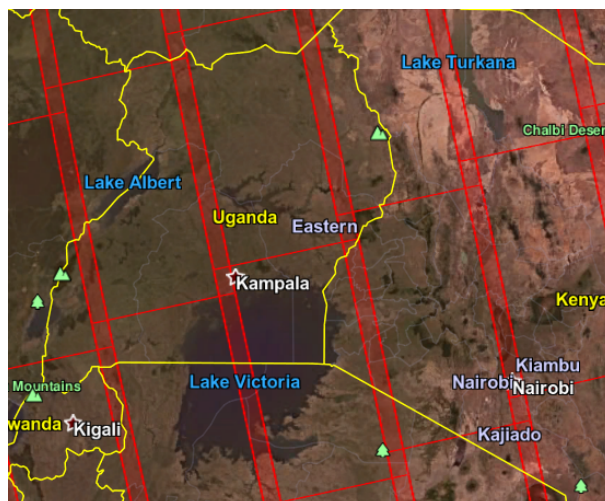
## Teoreetilised alused

### 1.1 Sentinel satelliidid

Aastal 2014 jõudis orbiidile Euroopa Liidu satelliit Sentinel-1A, mis on osa Copernicuse programmist, mille raames saadetakse orbiidile uus perekond Sentinel kaugseiresatelliite. Iga Sentineli missioon koosneb kahest satelliidist, millega tagatakse vajalikud katvused ning ülelendude sagedused.

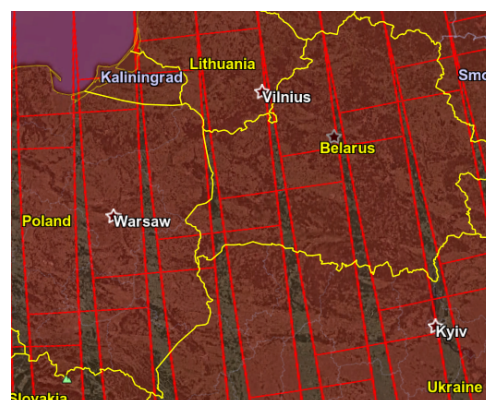
Kõrgeima ruumilise lahutusega Copernicus programmi satelliidid on Sentinel-1 ja Sentinel-2. Sentinel-1 kannab tehisavaradarit (ingl. k. Synthetic Aperture Radar - SAR), mis töötab C-riba mikrolainetel ehk keskmise sagedusega 5,405 GHz. Sentinel-1 kogub SAR andmeid nii öösel kui päeval ning pea kõikides ilmaoludes. Sentinel-2 on mitmespektriline kõrge lahutusega satelliit, mille optilisel sensoril on 13 spektraalkanalit. Neli neist on nähtava ja lähisinfrapuna lainepikkuste vahemikus ruumilise lahutusega 10 m, kuus kanalit on infrapuna ja 700 nm - 2400 nm vahemikus lahutusega 20 m ja kolm vahemikku on atmosfäärilise parandi kanalid lahutusega 60 m. [26],[27]

Sentinel-1 koosneb kahest satelliidist Sentinel-1A ja Sentinel-1B, mis kumbki teevad Maast iga 12 päevaga nii tõusva kui langeva orbiidi ülesvõtte. Ehk ekvaatoril on 6 päeva jooksul kaks ülelendu, näiteks Sentinel-1A tõusvad orbiidid pakuvad ekvaatoril joonisel 1.1 kujutatud katvust:



Joonis 1.1: Sentinel-1 tõusvate ülelendude potentsiaalne ülesvõtte pindala ekvaatoril, vaikerežiimi IW korral.

Sentinel-1 piltide ülesvõtte laius on konstante, seega laiuskraadi suurenedes hakkab polaarorbiidi korral erinevate ülelendude omavaheline kattuvus suurenema. Vt. joonis 1.2



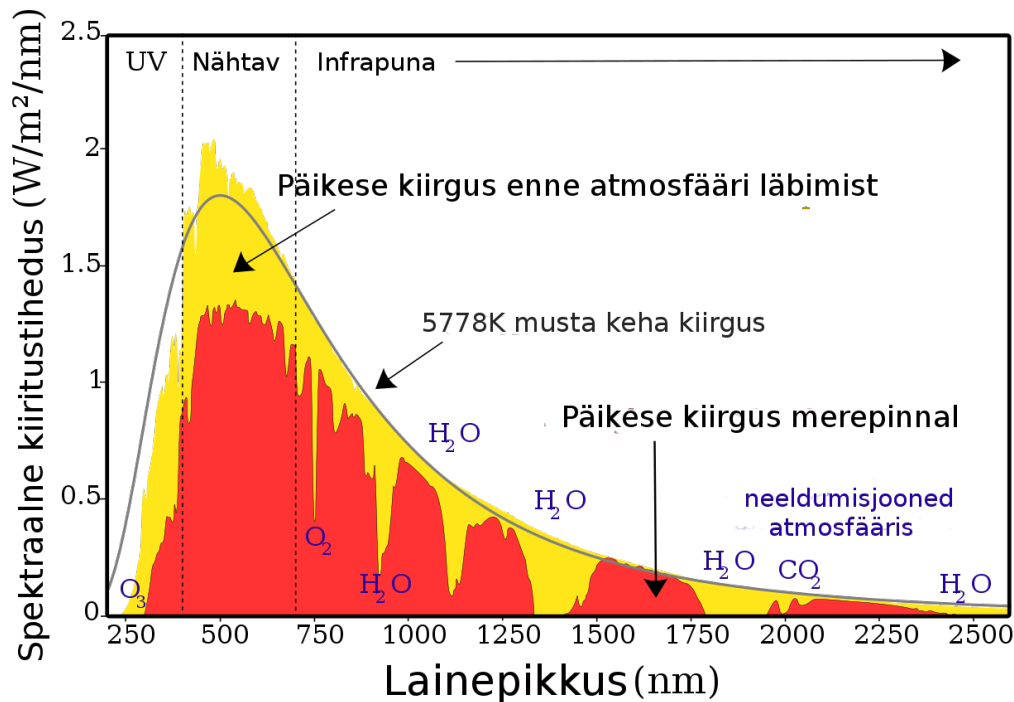
Joonis 1.2: Sentinel-1 tõusvate ülelendude potentsiaalne ülesvõtte pindala laiuskraadidel 37,9 (vasakul) ja 52,2 (paremal).

## 1.2 Optiline kaugseire

Optiliseks kaugseireks nimetatakse eemal asuvate objektide kohta informatsiooni hankimist optiliste mittekontaktsete meetoditega. Käesolevas töös käsitleme ainult Maa spektromeetrilist uurimist Sentinel-2 satelliidiga.

### 1.2.1 Valguse vastasmõju ainega

Passiivses optilises kaugseires on valgusallikaks Päike, mida saab heas lähenduses käsitleda mustakeha kiirgurina, mille temperatuur on 5778 Kelvinit. Joonisel 1.3 on näha, kuidas Päikeselt lähtuva valguse spekter muutub läbides Maa atmosfääri.



Joonis 1.3: Päikeselt lähtuv kiiritustiheduse spekter ning meretasemeni jõudva valguse kiiritustiheduse spekter [28]. Valguse neeldumine ning hajumine atmosfääris olevates erinevates osakestes tekitab päikese valguse spektrisse lainepikkuse piirkondi, kus enamus valgust ei jõua meretasemeni.

Sattudes vastasmõjusse ainega valgus kas neeldub, tungib ainst läbi, hajub või peegeldab. Erinevad materjalid peegeldavad ja neelavad valgust erinevalt sõltuvalt valguse lainepikkusest. Materjali kiirgusspekter iseloomustab kiirguse võimsuse jaotust lainepikkuse või mõne muu spektraalmuutuja järgi. Põhimõtteliselt on võimalik materjali tuvastada tulenevalt selle materjali kiirgus- või muust spektrist, eeldades piisavalt suurt spektraalset lahutusvõimet, et eristada selle materjali spektrit teistest materjalidest. Sellel põhimõttel põhineb optiline multispektraalne kaugseire. [29]

### 1.2.2 Kaugseire mõõtetulemusi iseloomustavad suurused

Kaugseire mõõtmisi iseloomustatakse ruumilise, spektraalse, radiomeetrilise ning ajalise lahutusvõimega:

- Ruumiline lahutusvõime on seotud ühele pildi pikslile vastava ruudukujulise alaga. Tavaliselt ruumiline lahutus on suurusjärgus 0,1 kuni 1000 meetrit.
- Spektraalne lahutusvõime määrab mõõdetava valguse lainepikkuste eristatavuse. Sentinel-2 ribasid on 13 tükki ning nende laiused varieeruvad 18 nanomeetrist 242 nanomeetrini.
- Radiomeetriline lahutusvõime määrab mitut erinevat kiirgusintensiivsuse taset sensor eristada suudab. Tavaliselt see väärtus on 8 ja 14 biti vahemikus, ehk 256 erinevast tasemest kuni 16384 erineva tasemeni.
- Ajaline lahutusvõime tuleb satelliidi ülelendude sagedusest ning on tähtis uurimustes, kus analüüsitakse satelliidipiltide aegrida.

Edukaks kaugseire mõõtmiseks on tähtis, et kaugseirega mõõdetav objekt oleks samas suurusjärgus ruumilise lahutusvõimega. Näiteks üksik põllutaim pole satelliidipildilt eristatav, kuid homogeenne põllutäis samasuguseid taimi on eristatav. Probleeme võivad tekitada põldude piirid, kus ühele satelliidipildi pikslile jääb osaliselt põld ning osaliselt midagi muud.

Sentinel-2 piltide pikslite väärtused on 12-bitilised ning on otseselt võrdelised kirkusega. [30]

Kirkus (ingl. k. radiance) on radiomeetriline suurus, mis iseloomustab ruumi- ja mittepunktvalgusallikate kiiratud valgust. Kirkuse ühikuks on  $\frac{W}{m^2 \cdot sr}$  ning see kirjeldab kiirgusvoogu ruuminurga ja kiirgust väljasaatva efektiivpinna kohta.

Kirkuse valem on:

$$L_e = \frac{d\Phi_e}{dA \cos(\theta) d\Omega} \quad (1.1)$$

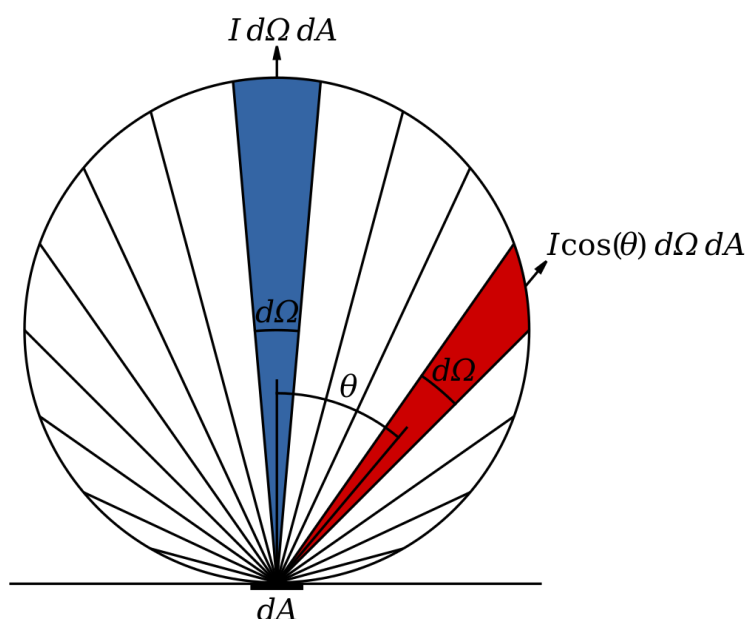
Kus  $dA$  on nähtava pinna pindala,  $\theta$  on nurk pinnanormaali ning vaatlemise sihi vahel,  $d\Phi_e$  on kiirgusvoog ning  $d\Omega$  on kiirgusvoo suunaline ruuminurk.

Kui tegu on ideaalselt hajutava kehaga, mis hajutab valgust igas sihis ühtlaselt, kehtib Lamberti seadus. Lamberti seadus väidab, et pinnalt hajunud kiirgustugevus on võrdeline pinnalt hajunud valguse langemisnurga ning pinnanormaali vahelise nurga koosinusega:

[31]

$$I(\theta) = I \cos(\theta) d\Omega dA \quad (1.2)$$

Joonisel 1.4 on kujutatud ideaalselt hajutavat keha, millelt peegelduvat valgust kirjeldab Lamberti seadus.



Joonis 1.4: Valgustugevused normaaltasandis ning  $\theta$  kraadi kõrvale kallutatult. Valgustugevus iga kiilu suunas on võrdeline kiilu pindalaga.[32]

Kui Sentinel-2 piltide pikslid oleks väljendatud kiirgustugevuses, siis oleks satelliidipildi äärmised pikslid Lamberti seaduse tõttu erinevate väärtustega. Kirkus aga ei sõltu pealelangeva valguse nurgast, mistõttu on Sentinel-2 pildid väljendatud kirkuses.

Sentinel-1 tööpõhimõtete kohta on võimalik lugeda lisast A.1.

### 1.2.3 Optilises kaugseires mõõdetavad lainepikkuste vahemikud

Optilises kaugseires kasutatavad lainepikkuste ribad on väga hoolikalt valitud. Näiteks Sentinel-2 lähisinfrapuna riba 8a on vaid 33 nm lai, keskmise lainepikkusega 864,7 nm, selleks et olla tundlik taimestiku lähisinfrapuna kiirgusele ja mullastiku raudoksiidi sisaldusele, kuid hoiduda veeauru mõjudest. [33]

Järgnevalt loetletakse tähtsaimad lainepikkuste vahemikud optiliseks kaugseireks, koos nende rakendustega:

- 1. Mereline aerosool (430 nm-450 nm). Kuna 430 nm-450 nm riba valgust neelab vesi vähem, siis kasutatakse seda rannikuvete, ookeanite värvuse ja mere taimestiku jälgimiseks. Lisaks mängib see riba suurt rolli pilvede, udu ja suitsu tuvastamises, et pilvi paremini välja filtreerida. Vastab Sentinel-2 kanalile 1.

- 2. Sinine (450 nm-510 nm). Sügavate vete uurimiseks sobib sinine riba hästi, sest sinine valgus neeldub vees kõige vähem. Lisaks on Rayleigh hajumise tõttu sinine lainepikkus väga tundlik pilvedele, udule ja suitsule, mistõttu kasutatakse seda riba nende tuvastamiseks ja filtreerimiseks. Vastab Sentinel-2 kanalile 2.
- 3. Roheline (530 nm-590 nm). Roheline riba katab peegelduse maksimumi roheliste lehtede pindadelt, kus on lisaks sinine ja punane peegeldustegur väga madal, kuna fotosünteesiks neid lainepikkusi neelatakse. Rohelist riba kasutatakse taimse materjali tuvastamiseks, vetikate ja tsüanobakterite vohamise otsimiseks. Vastab Sentinel-2 kanalile 3.
- 4. Kollane (585 nm-625 nm). Kollast riba kasutatakse kollaste objektide leidmiseks. Näiteks haigusega nakatunud taimed värvuvad kollakaks ning paljudel taimedel on õitsedes kollaseid taimeosi. Puudub Sentinel-2 kanalitest.
- 5. Punane (640 nm-670 nm). Kasutatakse taimede tuvastamiseks, sest taimeistik neelab punast valgust. Veel uuritakse punase ribaga ka mullatüüpe ning geoloogilisi objekte, sest raudoksiidi sisaldavad mullad on punased. Punast riba kasutatakse NDVI arvutamisel (ingl. k. Normalized Difference Vegetation Index). Vastab Sentinel-2 kanalile 4.
- 6. Punase serv (ingl. k. red-edge) (705 nm-745 nm). Klorofüll neelab seda lainepikkuse vahemikku tugevalt. Kasutatakse taimede vanuse ning elujõulisuse hindamiseks. Punase serv on tihti parim lähendus klorofüllis sisalduse ning lehtede pindala hindamiseks. Veel saab punase serva riba rakendada tervete ning haigete põllukultuuride eristamiseks. Vastab Sentinel-2 kanalitele 5 ja 6.
- 7. Lähinfrapuna 1 (760 nm-900 nm). Selle ribaga on parim liigitada taimeistikku elujõuliseks ja mitte elujõuliseks. Lähinfrapunas on kerge eristada vett ning taimeistikku, kuna terved taimed peegeldavad lähinfrapuna, kui samaaegselt vesi pigem neelab infrapuna. Veel hinnatakse selle riba abil biomassi, sest terves seisundis klorofüll peegeldab lähinfrapuna ning taimede surres hääbub lähinfrapuna peegeldumine. Lähinfrapuna 1 lainepikkuste vahemikku kasutatakse NDVI-s, et hinnata taimede elujõulisust, sest elujõuliste lehtede sisemine struktuur on väga heaks lähinfrapuna hajutavaks keskkonnaks. Vastab Sentinel-2 kanalitele 7, 8 ja 8A.
- 8. Lähinfrapuna 2 (860 nm-1040 nm). Seda riba mõjutavad atmosfääri mõjutused vähem kui lähinfrapuna 1-te, mistõttu on see lainepikkuste vahemik lähinfrapuna 1-st paremini kasutatav taimeistiku ning veepiiride uurimiseks. Vastab Sentinel-2 kanalitele 8 ja 8A.
- 9. Kiudpilvede riba (1360 nm-1380 nm). Selle riba nimi pärineb edukast kiudpilvede tuvastamisest. Atmosfäär neelab peaaegu täielikult selle lainepikkuste vahemiku, kuid see riba võimaldab atmosfääri kõrgetes kihtides olevaid pilvi tuvastada. Vastab Sentinel-2



kanalile 10.

- 10. Lühilaineline infrapuna 1 (ingl. k. Short Wave Infrared 1 ehk SWIR 1) (1570 nm-1650 nm). See riba on tundlik niiskuse sisaldusele mullastikus ning taimestik, kuna nendel lainepikkustel peegeldustegur väheneb veesisalduse suurenedes. Lisaks on SWIR 1 tuntud võime poolest läbida õhukesi pilvi. Vastab Sentinel-2 kanalile 11.
- 11. Lühilaineline infrapuna 2 (ingl. k. SWIR 2) (2080 nm-2350 nm). SWIR 2 on sarnane SWIR 1-ga, kuid SWIR 2-te kasutatakse peamiselt mullatüüpide ja teatud savitüüpide uurimiseks. Lisaks on selle lainepikkuste vahemikuga hea lahti mõtestada niisutamisi, sest SWIR 2 võimaldab hästi kaardistada veepuuduses olevate taimedega piirkondi. Vastab Sentinel-2 kanalile 12.
- 12. Soojuslik infrapuna (10600 nm-12510 nm). Nendel lainepikkustel näeb riba soojust, mis võimaldab hästi jälgida näiteks linnade soojust, vulkaanide tegevust ning kasutada sisendina ilmaennustuses. Puudub Sentinel-2 kanalitest.

[34]

## 1.2.4 Vegetatsiooniindeksid

Vegetatsiooniindeksiks nimetatakse igat suurust, mis on arvutatud kirkuse andmetest, mis on seotud taimede kogusega piirkonnas, nagu näiteks lehtede pindala või protsent pinnasest.

Kõige levinum vegetatsiooniindeks on NDVI ehk normeeritud vahe vegetatsiooni indeks (ingl. k. Normalized Difference Vegetation Index):

$$NDVI = \frac{(NIR - Punane)}{(NIR + Punane)} \quad (1.3)$$

NDVI töötab, sest elujõulised rohelised taimed neelavad päikesevalgust fotosünteesiliselt aktiivses kiirguse piirkonnas, et fotosünteesiga energiat toota. Taimede lehtede rakud on arenenud, et tagasi kiirata energiat, mis on umbes pool vastuvõetud päikese energiast, sest alles jäänud rohkem kui 700 nm lainepikkusega footonitel pole piisavalt energiat orgaaniliste molekulide sünteesimiseks. Ilma lähiinfrapuna valgust kiirgamata kuumeneksid taimed üle ja tõenäoliselt kahjustaks oma kudesid. Seega näevad rohelised taimed fotosünteesiliselt aktiivses kiirguse piirkonnas välja tumedad ning lähiinfrapunas eredad.

Veel näiteid vegetatsiooni indeksitest on näiteks lehepinnaindeks ehk LAI (ingl. k. Leaf Area Index) ning EVI (ingl. k. Enhanced Vegetation Index). [31]

NDVI-d kasutatakse tihti teiste vegetatsiooniindeksite ja taimede omaduste hindamiseks. Näiteks LAI, biomassi ja klorofüllü kontsentratsiooni lehtedes ja taimede produktiivsust. Selline lähendamine on võimalik kombineerides NDVI väärtusi kohapeal tehtud täiendavate mõõtmistega. Tulemusena tekib aga probleeme mitme mõõtmise ühildamisega, sest kohapealne mõõtmine on tihti kõrgema ruumilise lahutusvõimega kui satelliidipilt. [35]

## 1.3 Masinõpe

Masinõpe on teadusharu, mis uurib algoritme ja statistilisi mudeleid, mida kasutatakse ülesannete tõhusamaks lahendamiseks ilma konkreetsete juhusteta, tuginedes hoopis andmetest mustrite leidmisele ning andmestruktuuride põhjal järelduste tegemisele. Masinõpe jaguneb kolmeks peamiseks suunaks:

- Juhendatud õpe (ingl. k. supervised learning). Algoritmile antakse treenimiseks näidissisendid ning igale näidisele vastavad soovitud väljundid ehk sildid (ingl k. labels). Eesmärk on leida funktsioon, mis suudab igale sisendile määrata vastava väljundi.
- Stiimulõpe (ingl. k. reinforcement learning). Algoritmile antakse tagasisidet preemia või karistusena vastavalt algoritmi tegudele dünaamilises keskkonnas nagu auto juhtimine või mängu mängimine vastase vastu.
- Juhendamata õpe (ingl. k. unsupervised learning). Algoritm peab ise leidma sisenditest struktuure. Juhendamata õpet kasutatakse tihti, mustrite ja klastrite leidmiseks paljumõõtmelises andmehulgas. Ühtegi sildistatud sisendit ei anta masinõppe algoritmile. [36]

lahendada Üldiselt kasutatakse loetletud suundades väga erinevaid algoritme, kuid mõningate keerulisemate ülesannete lahendamiseks masinõppe liike kombineeritakse. Näiteks isesõitvates autodes leidub nii juhendatud kui ka stiimulõppe närvivõrke, mis koos paljude lihtsamate algoritmidega loovad väga võimeka tehisintellekti. Selle uurimuse raames käsitletakse ainult juhendatud õppe algoritme.

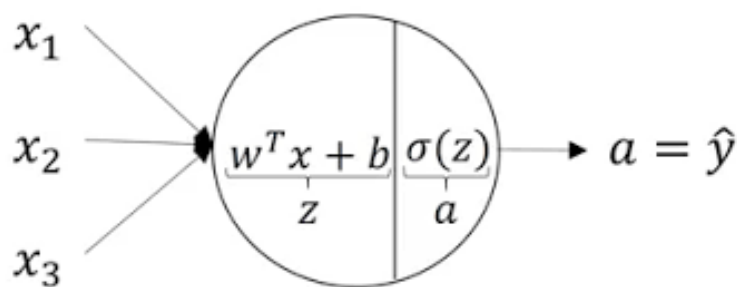
## 1.4 Tehisnärvivõrgud

Tehisnärvivõrgud on bioloogilise aju eeskujul ehitatud masinõppe arhitektuurid, mis võimaldavad automatiseerida ülesandeid või teisiti öelduna, luua tehisintellekti. (Tehis)närvivõrgud on masinõppe alamharu ning neid kasutatakse kõigis masinõppe suundades.

### 1.4.1 Logistiline regressioon

Kõige lihtsamaks masinõppe närvivõrguks on logistiline regressioon. Samaaegselt on logistiline regressioon lihtsaim masinõppe sõlm. Sõlmesid tähistatakse ringidega ning iga sõlm rakendab teatavat matemaatilist operatsiooni kõikidele sõlme sisenditele.

Olgu sisenditeks  $x_1, x_2$  ja  $x_3$  ning igale sisendile vastav kaal. Kokku on sõlmel 3 kaalu  $\omega_1, \omega_2$  ja  $\omega_3$  ning üks vabaliige  $b$ . Maatrikskorrutamine liidab kõikide sisendite ja kaalude korrutise kokku ning lisab vabakaalu  $b$ , tulemuseks saadakse suurus nimega aktivatsioon  $z$ . Aktivatsioonist saab aktivatsioonifunktsiooni  $\sigma$  argument. Väljundiks on masinõppe algoritmi ennustus.



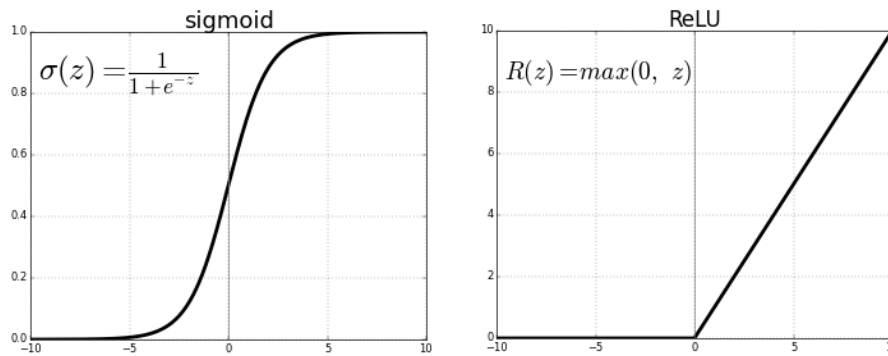
Joonis 1.5: Lihtsaim masinõppe närvivõrk - logistiline regressioon. [37]

### 1.4.2 Aktivatsioonifunktsioon

Aktivatsioonifunktsioon defineerib sõlme väljundväärtuse vastavalt sõlme sisenditele. Aktivatsioonifunktsiooni väljundist saab järgmise sõlme sisend. Aktivatsioonifunktsioone on vaja, et tuua närvivõrku mittelineaarsust. Kui poleks mittelineaarseid aktivatsioonifunktsioone, ei suudaks närvivõrk tekitada sirgetest erinevaid otsustuspiire ning kõik närvivõrgu kihid oleks võimalik taandada üheks lineaarseks teisendust tegevaks kihiks. [37]

Ajalooliselt kasutati aktivatsioonifunktsioonina peamiselt sigmoid funktsiooni, mis normeerib sisendit nii, et väljundväärtus on alati 0 ja 1 vahel. Viimase sõlme väljundi binaarseks liigitamiseks kasutatakse sigmoid funktsiooni tänaseni.

Närvivõrgu vahekihtides eelistatakse tänapäeval sigmoid funktsiooni asemel kasutada ReLU funktsiooni, mis on arvutuslikult kiirem ning hoiab ära probleemi nullilähedase tuletise tekkimisega, mis muudab närvivõrgu treenimise väga aeglaseks.



Joonis 1.6: Sigmoid ja reLU aktivatsioonifunktsioonid koos valemitega. [38]

Tulemi mitmesse klassi liigitamiseks kasutatakse käesoleva töö närvivõrgu viimases kihis softmax aktivatsioonifunktsiooni:

$$p_i = \frac{e^{s_i}}{\sum_j^C e^{s_j}} \quad (1.4)$$

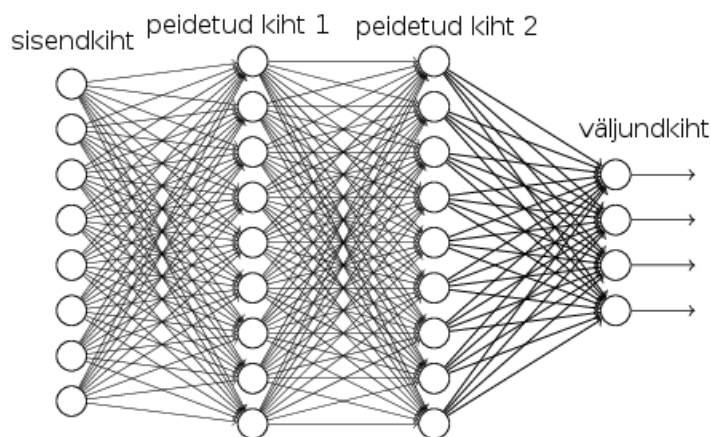
Kus  $s_j$  on igale klassile vastav sisendväärtus viimases kihis.  $C$  on klasside arv.  $p_i$  on tõenäosus, et sisend kuulub  $i$ -ndasse klassi.

### 1.4.3 Süvaõppe närvivõrgud

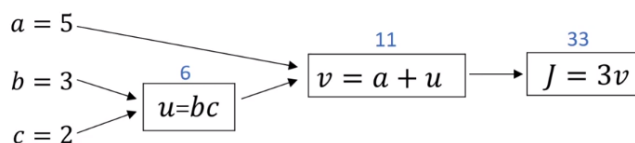
Süvaõppe algoritmiks nimetatakse närvivõrku, mis on vähemalt kaks või rohkem peidetud kihti sügav [36]. Sisend- ja väljundkihte ei loeta peidetud kihtideks. Joonisel 1.2 kujutatud ringe nimetatakse sõlmedeks ning iga sõlm teostab teatavat matemaatilist operatsiooni, näiteks esmalt tehakse logistilist regressiooni, pärast mida kasutatakse samas sõlmes aktivatsioonifunktsioonina ReLU.

### 1.4.4 Pärilevi

Pärilevi (ingl. k. Forward Propagation) korral liiguvad närvivõrgu sisendid edasi mööda sõlmesid, kuni saadakse närvivõrgu väljund. Pärilevi samm-sammult jälgimiseks on mõistlik kirjutada arvutuslik graaf (ingl. k. computational graph), kus on näha sisenditega tehtud operatsioonid. Pärilevi tulemusena saadakse väljundi arvuline väärtus, mida on vaja kaofunktsiooni leidmiseks (ingl. k. loss function).



Joonis 1.7: Lihtsaim süvaõppe närvivõrk, mis koosneb sisend- ja väljundkihist ning kahest peidetud kihist



Joonis 1.8: Närvivõrgu osad tehakse arvutuslikuks graafiks, et pärast oleks kergem leida kaofunktsiooni tuletist iga sisendmuutuja järgi.  $u$ ,  $v$  ning  $J$  tähistavad sõlmedele vastavaid operatsioone,  $a$ ,  $b$ ,  $c$  on sisendkihi muutujad. [37]

### 1.4.5 Kaofunktsioon

Kaofunktsiooni kasutatakse närvivõrgu vea hindamiseks. Kaofunktsioon arvutatakse iga sisendi jaoks eraldi ning kaofunktsiooni lihtsaimaks näiteks on binaarse klassifikatsiooni korral kasutatav ristentroopia kaofunktsioon:

$$L(\hat{y}, y) = -y \ln \hat{y} - (1 - y) \ln(1 - \hat{y}) \quad (1.5)$$

$\hat{y}$  on ennustatud sildi tõenäosus vahemikus 0 kuni 1 ja  $y$  on sisendile vastav tegelik silt, 0 või 1.

Käesolevas töös tuleb väljundid jaotada mitmetesse klassidesse, seega kasutatakse kategoorilist ristentroopia kaofunktsiooni softmax aktivatsioonifunktsiooniga.

$$L(p, y) = - \sum_i y \ln p_i \quad (1.6)$$

$y$  on klassi tegelik sildi väärtus.  $p_i$  on  $i$ -nda klassi tõenäosus softmax aktivatsioonifunktsioonist.  $L$  on kategooriline ristentroopia.

### 1.4.6 Kulufunktsioon

Kui kaofunktsioon on arvutatakse iga sisendi kohta, siis kulufunktsioon on kõikide kaofunktsioonide aritmeetiline keskmine:

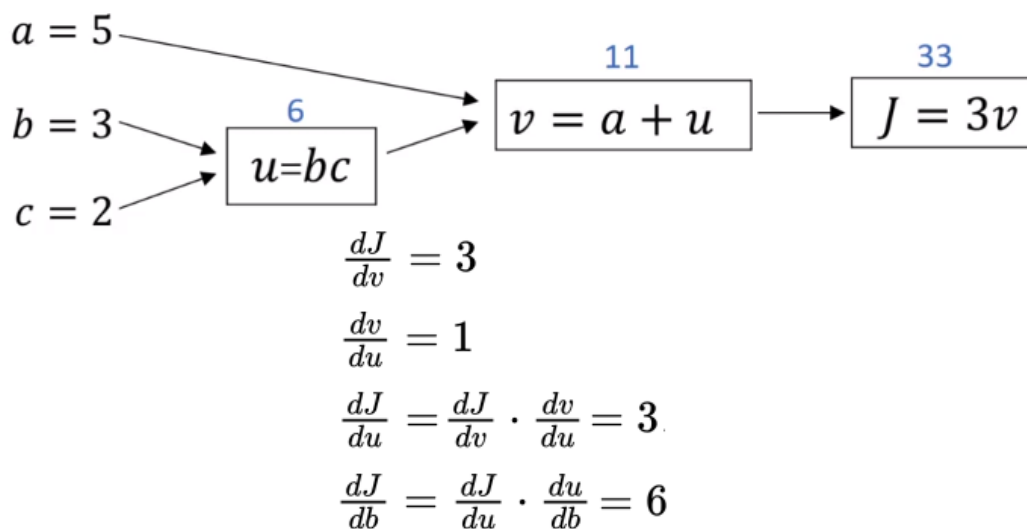
$$J(L) = \frac{1}{m} \sum_i^m L(p, y) \quad (1.7)$$

, kus  $m$  on algoritmi sisendite arv.

Algoritmi treenimise eesmärk on viia kulufunktsioon miinimumi. Selleks kasutatakse tagasilevi (ingl. k. backpropagation) ning gradientlaskumist.

### 1.4.7 Tagasilevi

Tagasilevis liigutakse arvutusgraafis tagasi - väljundist sisendite suunas. Kaofunktsioonide tulemuste leidmiseks iga sisendmuutuja suhtes rakendatakse ahelreeglit.



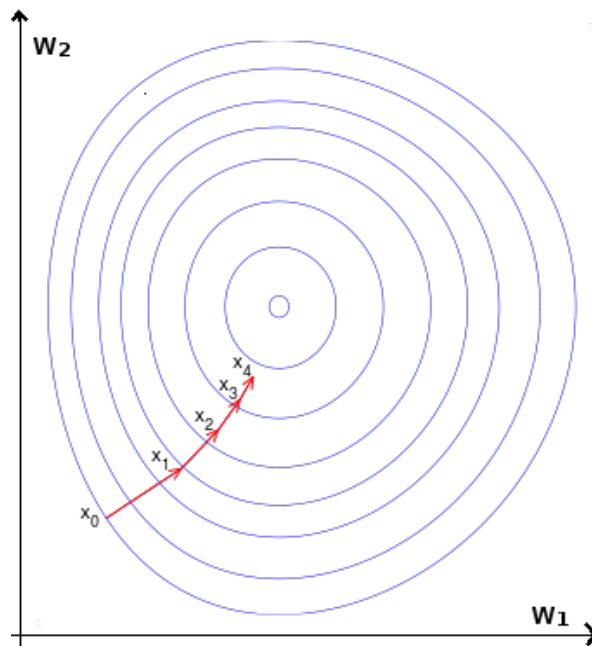
Joonis 1.9: Tagasilevi käigus kasutatakse ahelreeglit, et leida kaofunktsiooni tulemised iga sisendmuutuja järgi. Joonisel leiti kaofunktsiooni tulemus sisendmuutuja  $b$  järgi.

### 1.4.8 Gradientlaskumine

Kulufunktsiooni tuletise saab leida kaofunktsiooni tuletiste kaudu:

$$\frac{\partial}{\partial \omega_i} J(\omega, b) = \frac{1}{m} \sum_{i=1}^m \frac{\partial}{\partial \omega_i} L(a^{(i)}, y^{(j)}) \quad (1.8)$$

Teades kulufunktsiooni osatuletisi kõikide sisendmuutujate järgi, saame neid kokku summeerides gradiendi, mis näitab kulufunktsiooni kõige järsema languse suunda. Vt. joonis 1.10:



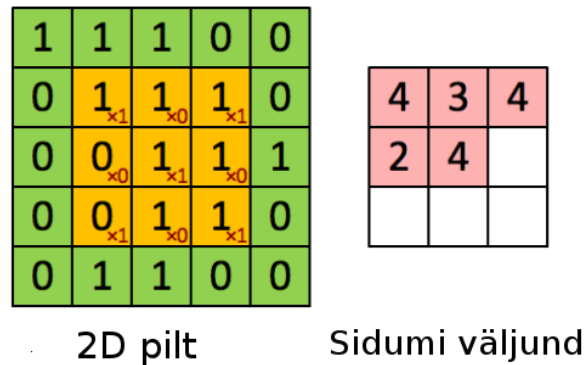
Joonis 1.10: Gradientlaskumist on kõige lihtsam visualiseerida kulufunktsioonil, mis sõltub kahest sisendmuutujast. Gradientlaskumise meetodil muudame sisendmuutujaid nii, et liigume kulufunktsiooni globaalse miinimumi suunas kõige kiiremini. [37]

## 1.5 Sidumnärvivõrk ehk CNN

Sidumnärvivõrk (ingl. k. Convolutional Neural Network ehk CNN) on levinuim närvivõrguarhitektuur, mida kasutatakse nii piltidelt objektide tuvastamiseks kui ka piltide semantiliseks segmenteerimiseks ehk pildi osade klassidesse jagamiseks.

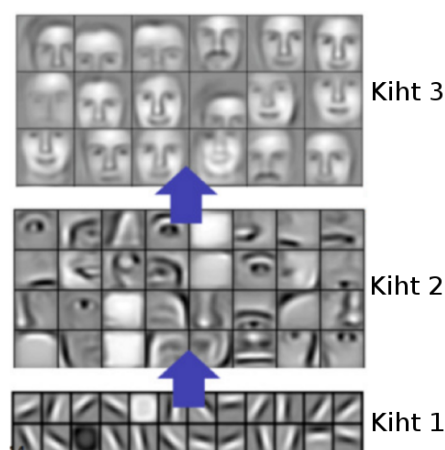
### 1.5.1 Sidumoperatsioon ja sidumkiht

Joonis 1.11 kujutab 2D sidumoperatsiooni, kus sisendiks on kahemõõtmeline pilt ning väljundiks vähendatud mõõtmetega kahemõõtmeline pilt.



Joonis 1.11: Sidumnärvivõrgu peamine operatsioon on sidumi võtmine filtritega. Joonisel oleva oranži ruuduga on toodud filtri asukoht ning oranži ruudu elementide sees olevad punased väärtused on filtri õpitavad kaalud, mis määratakse gradientlaskumisega, minimeerides kulufunktsiooni. Kõik filtri ja pildi elementide korrutised liidetakse ning saadakse sidumi väljundmaatriksi üks element.

Sidumnärvivõrgu iseloomulikum osa on sidumoperatsiooni teostav kiht. Joonisel 1.12 on näha kolme erineva sidumkihi filtrite kaalude kujutised sidumnärvivõrgus, mida rakendati näopiltide nädisandmekogu peal. Jooniselt nähtub, et filtrid võtavad teatud üldistatuse tasemega nädisandmete kuju.



Joonis 1.12: Sidumnärvivõrgu esimestes kihtides leitakse servad ning teravad üleminekud, järgmistes kihtides hakatakse moodustama eelmiste kihtide põhjal keerulisemaid elemente nagu kõrvad, silmad, nägu. [18]



## 1.6 Tensorflow ja Keras

Käesolevas töös kasutatakse Tensorflow-d, mis on kõige laiemalt levinud masinõppe programmeerimise teek.

Keras on pythonis kirjutatud kõrge taseme rakendusliides närvivõrkude arendamiseks. Keras võimaldab sisemisel tasemel kasutada kas Tensorflow-d, CNTK-d või Theano-t. Keras võimaldab luua närvivõrkude mudeleid vähese koodi kirjutamisega, lubades uusi ideid kiiresti katsetada.

## Peatükk 2

# Kaasaegsed masinõppe arhitektuurid

Traditsioonilised masinõppe meetodid nagu otsustuspuude metsad (ingl. k. random forest ehk RF), tugivektorklassifitseerijad (SVM ehk ingl. k. Support Vector Machine) ja K-lähima naabri algoritm (KNN ingl. k. K-Nearest-Neighbours) jäävad süvaõppe närvivõrkude tulemustele täpsuses alla. [39], [40] Teisalt vajavad süvaõppe närvivõrgud võrreldes vanemate meetoditega suuremat andmekogu ning nende treenimine on arvutuslikult kallim. [41]

### 2.1 Ülevaateartiklid

Süveõppe kaugseire rakendustest kokkuvõtet pakkuvatest artiklitest oli kõige suurema kaetud artiklite arvuga Missisipi ülikooli J.E.Ball et al. [23]. Selles artiklis on kirjeldatud kõik peamised kaugseire uurimissuundade probleemid ning mitmesaja artikli pakutud lahendused on kompaktselt kokku võetud. J.E.Ball et al. artiklis käsitletakse ka mitme masinõppe süsteemi üheks kombineerimise küsimusi ning närvivõrkude arhitektuuride loomist, mis oleks invariantseid sisendandmete pöördele ning nihkele.

Hea ülevaate üldistest masinõppe uutest lähenemistest saab ka Massachusettsi Tehnoloogiainstituudi vanemteadur Lex Fridmani 2019 loengust. [42]

Üks parimaid klassikalise masinõppe meetodeid, millega tasub süvaõppe arhitektuure võrrelda, on XGBoost otsustuspuude mets. [43]

Hiljutised isesõitvate autode nägemise võistlused andmekogudel PASCAL VOC [44] ja Cityscapes [45] on võitnud CNN arhitektuur DeepLabV3+(Xception-JFT), mis põhinevad artiklitel [46] ja [47]. Neis artiklites on kasutatud vähe esinevate klasside klassifitseerimise täpsuse suurendamiseks alglaaduri (ingl. k. bootstrap) meetodit.

Samas peab pidama meeles, et süvaõppe arvutinägemise arhitektuurid pole üksüheselt üle kantavad kaugseiresse, kuna arvutinägemises on piltide lahutusvõime sadades kordades parem kui satelliidipiltide puhul. [48]

## 2.2 Masinõppe arhitektuuri valimine

Sisendpildi dimensioonidega klassifitseeritud väljundi saamiseks on peamiselt kaks meetodit: [49]

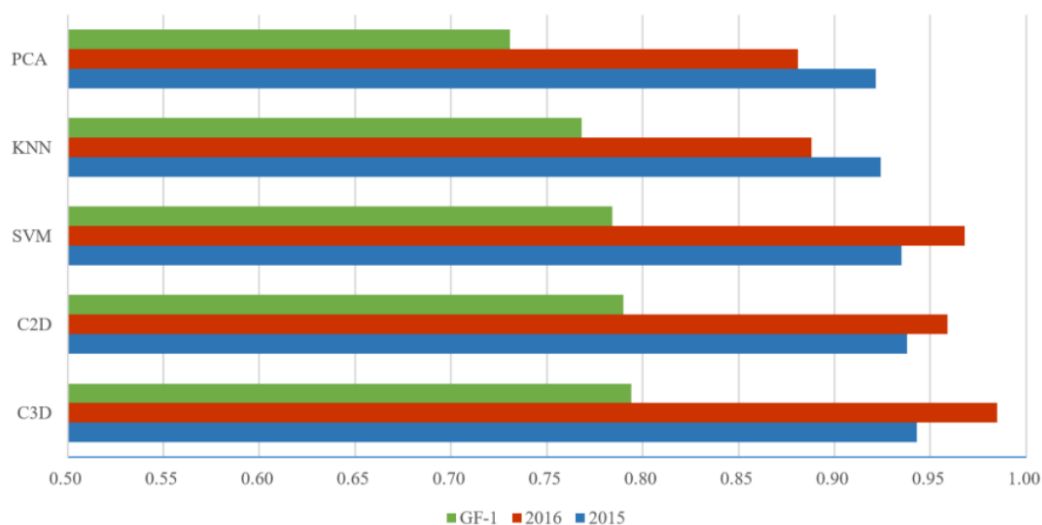
- Ümbruste meetod: Võtta sisendiks ühe piksli ümbrus ning väljundiks anda selle piksli klassifikatsioon. See tähendab, et ühe pildi klassifitseerimiseks tuleb algoritmi jooksutada pildi pikslite arv kordi. Selle lähenemise näiteks on artikkel [50].
- Täielik sidumnärvivõrk (ingl. k. Fully Convolutional Neural Network): Võtta sisendiks kogu pilt ning väljundiks anda sisendpildiga samade mõõtmetega pilt. See eeldab arhitektuuri erinevate astmete kaalude liitmist, kopeerimist või pöörd-sidumoperatsiooni tegemist, et enne lõppkihte aktivatsioonide mõõtmed taas suuremaks saada. Antud lähenemisel on mitmeid kombinatsioone, näiteks hüperveergude lähenemine [51], U-kujuline arhitektuur [49] või laienenud sidumi (ingl. k. Atrous Convolution) lähenemine [46].

Artiklis [52], vaadeldi põllukultuuride pikselhaaval klassifitseerimise täpsusi. Omavahel võrreldi 1D CNN-i, 2D CNN-i, ning ansambel-närvivõrku (ingl. k. Ensemble Neural Network ehk ENN) [53] ning otsustusmetsa (RF) kui tipptasemel klassikalise masinõppe esindajat. Tulemuseks saadi RF, ENN, 1D- ja 2D CNN täpsusteks vastavalt 88,7%, 92,7%, 93,5% ja 94,6%. Sidumnärvivõrgud olid kõige täpsemad ning 2D CNN edestas 1D CNN-i, kuigi mõningad väikesed objektid lõplikul klassifikatsiooni kaardil olid 2D CNN juhul silutud ning seetõttu valesti sildistatud. [52]

Artiklis [50] klassifitseeriti põllukultuure viite klassi ning leiti, et 3D CNNiga on võimalik saavutada täpsemaid tulemusi kui 2D CNNi ja tipptasemel traditsiooniliste masinõppe meetoditega nagu PCA(ingl. k. Principal Component Analysis), KNN ja SVM. Arhitektuuridega saavutatud täpsused on toodud joonisel 2.1:

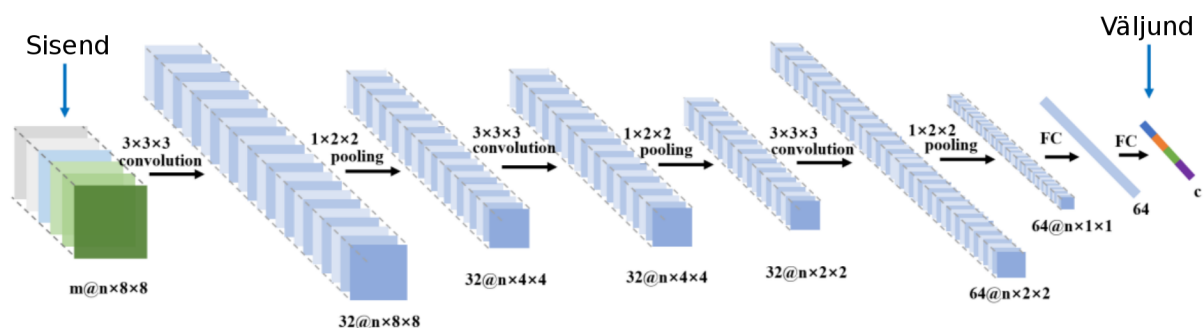
Kokkuvõtvalt saab eelmainitud artiklite tulemuste põhjal väita, et põllukultuuride klassifitseerimiseks on 3D CNN täpsem kui 2D CNN, mis on omakorda täpsem kui 1D CNN.

Täieliku sidumnärvivõrgu poole pealt on heaks näiteks viimase aasta autonägemise võistluse võitnud DeepLabV3+(Xception-JFT) [47]. Kuna täieliku sidumnärvivõrgu meetodi kohta leidub üks väga detailselt kirjeldatud Taani tehnikaülikooli magistritöö [54], kus uuriti Unet sidumnärvivõrku põllukultuuride liigitamiseks, siis otsustati käesolevas töös uurida näival



Joonis 2.1: Võrdlus klassikaliste masinõppemeetodite ja 2D ja 3D sidumnärvivõrkude klassifitseerimise kogutäpsuse vahel [50]. GF-1 on Gaofen-1 satelliidipiltidel põhinev mudel, kui 2015 ja 2016 on Gaofen-2 piltidel põhinevad mudelid.

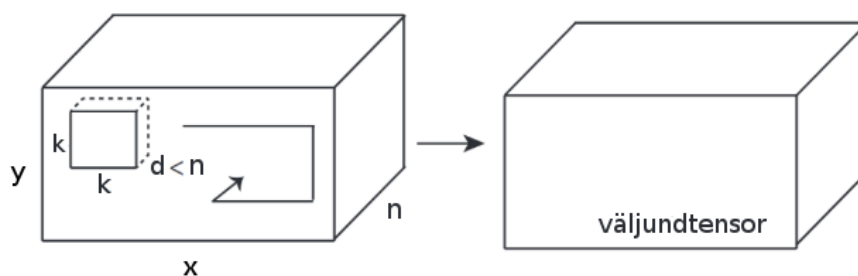
parimate tulemustega ümbruste meetodi närvivõrku, milleks on artiklis [50] kasutatud 3D sidumnärvivõrgu arhitektuur. Kasutatava arhitektuuri kihid on toodud joonisel 2.2.



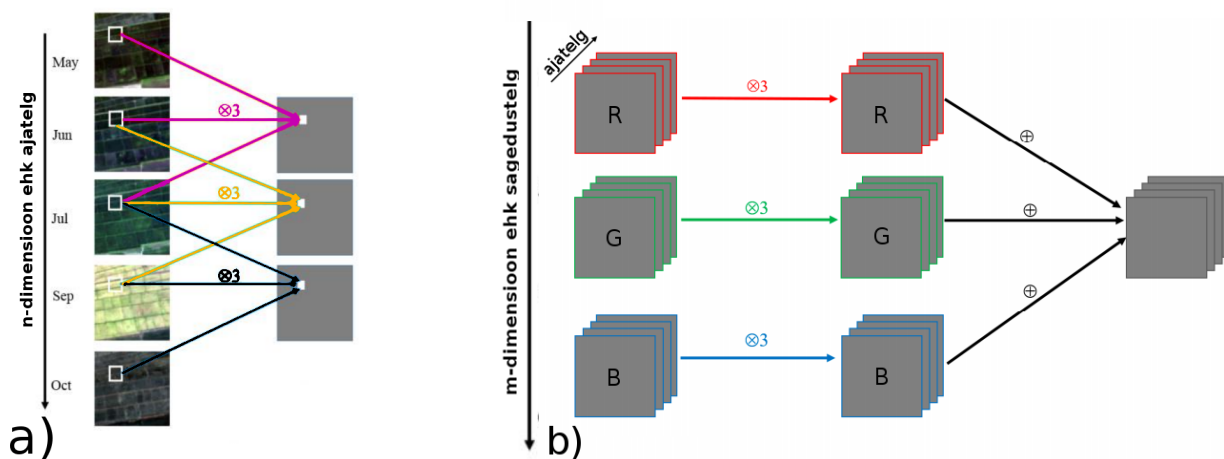
Joonis 2.2: Töös kasutatava 3D sidumnärvivõrgu arhitektuur, mis võtab sisendiks  $8 \times 8$  ümbruse ning annab väljundiks ühe piksli klassi.  $m$  on kanalite arv, tavaliselt (punane, roheline, sinine, NDVI),  $n$  on ajas tagasiulatuvate piltide arv ning  $c$  on klassifitseeritavate klasside arv.[50]

3D Sidumnärvivõrgu arhitektuuri tööst arusaamiseks peab täpsemalt mõistma 3D sidumoperatsiooni, mis on defineeritud järgmisel joonisel 2.3

Käesoleva töö sidumoperatsioonis rolli mängivad aja ning sagedusdimensioonid on täpsemalt kirjeldatud joonistega 2.4-a ja 2.4-b



Joonis 2.3: Esmalt tehakse 2D sidumoperatsioon x-y tasandil, siis sammutakse ühe sammu võrra edasi n-sihis (aeg) ning tehakse järgmine 2D sidumoperatsioon. Ladudes ritta 2D sidumoperatsiooni 2D väljundpilte, saadakse väljundtensor. [55]



Joonis 2.4: a) 3D sidumoperatsioon ajadimensioonis.  $\otimes 3$  tähistab 3D sidumoperaatorit, mis liidab treenitavate kaaludega kokku 3 erineva aja pilti. b) Erinevad kanalid, ehk erinevate lainepikkustega pildid, liidetakse treenitavate kaaludega kokku. [50]

# Peatükk 3

## Lähteandmed

### 3.1 Optimaalse uurimisala defineerimine

Hästi defineeritud uurimisala ja lähteandmed on olulised erinevate süvaõppe arhitektuuride objektiivseks võrdlemiseks.

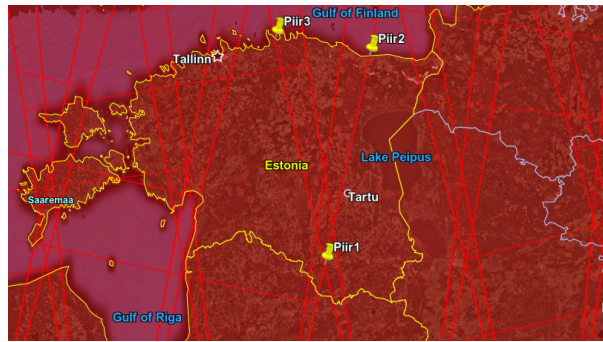
Kriteeriumid optimaalseks põllukultuuride klassifitseerimise algoritmi uurimisalaks on järgnevad:

- Ala peaks sisaldama võimalikult palju põlde
- Ala peaks täielikult sisalduma ühes Sentinel-1 ülelennus, et ei peaks uurimisala mitmest erineval ajal tehtud pildist kokku kombineerima
- Ala peaks täielikult sisalduma ühes Sentinel-2 ruudus

Uurimisala defineerimiseks kasutati Sentinel-1 orbiidi trajektoore KML vormingus (ingl. k. Keyhole Markup Language), Copernicus Open Access Hub-ist allalaetud Sentinel-2 pilti [56], Google Earth Pro-d ja QGISi. KML-id on KappaZeta otse küsinud ESA-lt (ingl. k. European Space Agency). QGIS on tasuta, avatud lähtekoodiga GIS (ehk Geograafiline Informatsiooni Süsteem), mis võimaldab sirvida ja töödelda vektor- ja rasterandmeid.

Eesti laiuskraadi juures on topelt kattuvus kõikjal, mis tähendab, et iga geograafilise punkti kohta on kuue päeva jooksul on saada vähemalt neli ülelendu, kaks tõusva ja kaks laskuva orbiidiga.

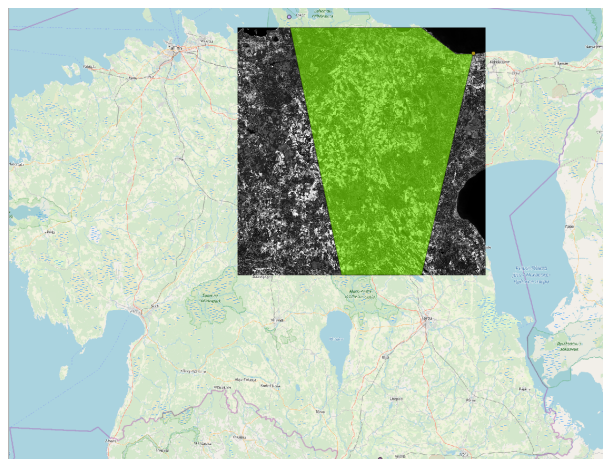
Joonisel 3.1 on kujutatud nii tõusvate kui ka langevate orbiitide kaadripiire Eesti kohal. Selleks, et kogu uurimisala neli pilti kuue päeva jooksul kuuluks samasse faili, tuleb piirduda kolmnurgaga, mis on toodud järgneval joonisel välja. Kui tahta võtta suuremat ala, mis võimaldaks igast geograafilisest punktist neli pilti, tuleks kombineerida rohkem kui nelja ülelennu pilte.



Joonis 3.1: Sentinel-1 tõusvate ja langevate orbiitide potentsiaalne ülesvõtte pindala Eestis

Ülal olevalt pildilt Google Earth-ist kopeeriti järgnevad koordinaadid QGIS-i: “Piir1”:  $57^{\circ}49'2,32''\text{N}$  ja  $26^{\circ}22'40,62''\text{E}$  “Piir2”:  $59^{\circ}26'4,69''\text{N}$  ja  $27^{\circ}4'48,85''\text{E}$  “Piir3”:  $59^{\circ}34'25,41''\text{N}$  ja  $25^{\circ}37'55,31''\text{E}$

Siinkohal on tähtis veenduda, et Google Earth ja QGIS mõlemad kasutaks õiget projektsiooni spetsiifilist koordinaatsüsteemi: “EPSG:3301 Estonian Coordinate System of 1997”. Seni, kuni käivad arvutused sfääril, on koordinaadid ühest süsteemist teise invariantse, kuid teisendades sfääril koordinaate 2D tasandile, tekivad ebatäpsused. Seetõttu on igal Maa piirkonnal oma koordinaatsüsteem, et minimeerida tasandile teisendamisel tekkivaid vigu. Sentinel-2 andmete ruut koos Sentinel-1 uurimisala piiridega on järgmine:



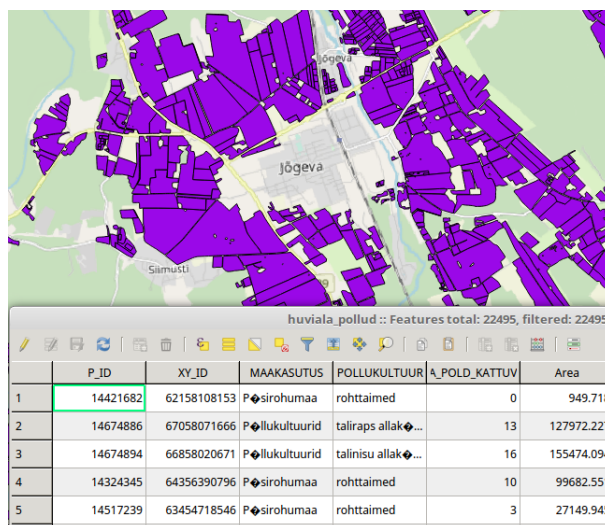
Joonis 3.2: Käesolevas töös kasutatav Sentinel-2 ruut idendifikaatoriga T32VMF Eesti kaardil, koos roheliseks värvitud optimaalse uurimisalaga, mis on kaetud parajasti nelja erineva Sentinel-1 orbiidi IW režiimi vaatealade ühisosaga, minimeerides vajamineva täieliku andmekomplekti hulka.

Kokkuvõttes: Uurimisala on piiritletud Sentinel-2 ruuduga, mille ID on T32VMF ning Sentinel-1 tõusvate orbiitide IW režiimi vaatealade ühisosaga, mille RON-id(ingl. k. Relative Orbit Number)

on 87 ja 160 ning langevate orbiitidega, mille RON-id on 80 ja 153.

## 3.2 Eristatavad põllukultuurid

QGIS-is joonestati uurimisala defineeriv vektorkiht, mille edastamisel PRIA-le, saadeti vastu uurimisalal olevate 22 495 põldude vektorkiht. Saadud vektorkiht sisaldab 2018. a. põllupiire koos igal põllul kasvava kultuuri nimega ning QGIS-i tarkvaraga arvutati ka põldude pindalad. Vt. joonis 3.3.



Joonis 3.3: Põldude vektorkiht koos atribuutide tabeliga, mis sisaldab iga põllu kohta põllul kasvavat kultuuri ning pindala.

PRIA andmebaasis on kõik Eesti põllukultuurid jaotatud ühte 484 liigitusest. PRIA huvi on tekitada põllukultuuride kaart, mis sisaldaks 27 põllukultuuri klassi, ehk koos mitte-põllukultuur klassiga on 28 klassi, mille vahel valida ühe piksli klassifitseerimisel.

Enamus klasse koosneb ühest põllukultuuri kombinatsioonist erinevate allakülvi variantidega. Näiteks 'Talitritkale' klassi kuuluvad liigitused: talitritkale allakülvita, talitritkale liblikõieliste allakülviga ja talitritkale allakülviga, v.a liblikõieliste allakülviga.

Teise seas on ka klass 'Muu', mis sisaldab palju erinevaid põllukultuure: hernes, arbuus, lehtkapsas, sinep, harilik pihlakas, võilill ning veel 300 liigitust. Klass 'Muu' loodi väikese külvipinnaga põllukultuuride kokku koondamiseks, et klassifitseerimisülesannet lihtsustada.

Lisaks on vahepealseid klasse nagu 'Heintaimed, kõrrelised', kuhu alla käivad: harilik ussikeel, kõrreliste rohumaa (vähemalt 80% kõrrelisi), muu heintaimede segu, rohttaimed, valkjäs



mesiohakas, üheaastane raihein jm üheaastased heintaimed, keerispea allakülviga, keerispea liblikõieliste allakülviga, keerispea allakülviga, v.a. liblikõieliste allakülviga

Eristatavate põllukultuuride klassid on toodud lisas A.3.

### **3.3 Projekti piiritlemine**

Arvutusressursi ja ajaliste piirangute tõttu kasutatakse käesolevas töös sisendandmetena ainult Sentinel-2 satelliidipilte.

Pilvevabade piltide valimiseks kasutati Sentinel Hubi tööriista [57]. Töös kasutati ainult selliseid pilte, mille pilvisus oli alla 5%.

Magistritöoga [54] võrreldavate tulemuste saamiseks valiti käesolevasse tööse, samamoodi nagu magistritöös, viis Sentinel-2 pilti maist:

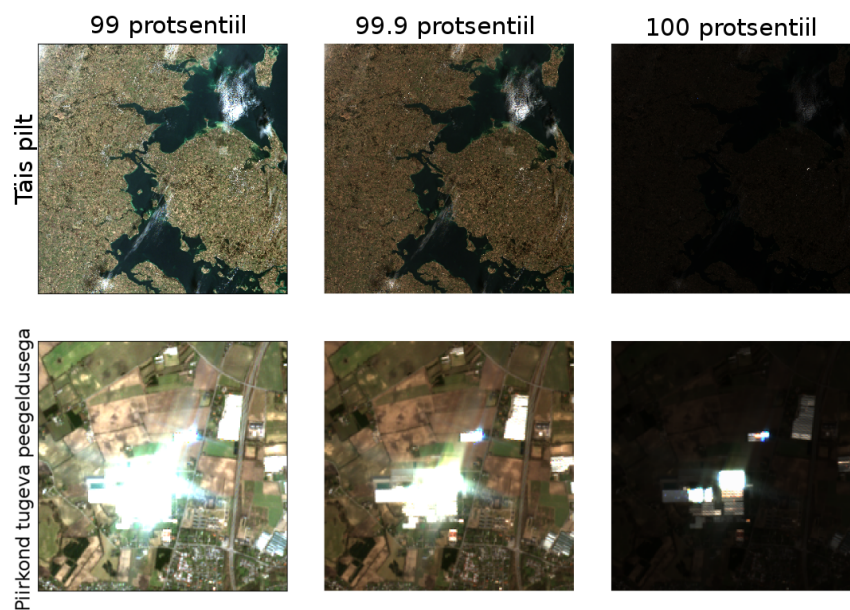
- S2A\_MSIL2A\_20180510T094031\_N0207\_R036\_T35VMF\_20180510T114819.SAFE
- S2B\_MSIL2A\_20180515T094029\_N0207\_R036\_T35VMF\_20180515T114639.SAFE
- S2A\_MSIL2A\_20180520T094031\_N0207\_R036\_T35VMF\_20180520T100921.SAFE
- S2B\_MSIL2A\_20180525T094029\_N0208\_R036\_T35VMF\_20180525T123136.SAFE
- S2A\_MSIL2A\_20180530T094031\_N0208\_R036\_T35VMF\_20180530T122531.SAFE

#### **3.3.1 Andmete normeerimine, peegelduste probleem**

Närvivõrkude treenimisel sisendandmed tavaliselt normeeritakse regulariseeriva ning treenimist kiirendava meetmena.

Mõnikord satub Sentinel-2 pildile aga otsepeegeldus, mis on normeerimise jaoks problemaatiline, nagu on näha joonisel 3.4.

Probleemi lahendamiseks jagati igit piksli heledust 99-protsentiiliga ning kõik väärtused piirati maksimaalselt üheni. Normeerimist teostav pythoni kood on toodud lisas A.5.



Joonis 3.4: Näide majadelt otse peegelduva valguse mõjust normeerimisele. [54]

### 3.4 Uute indeksite otsimine

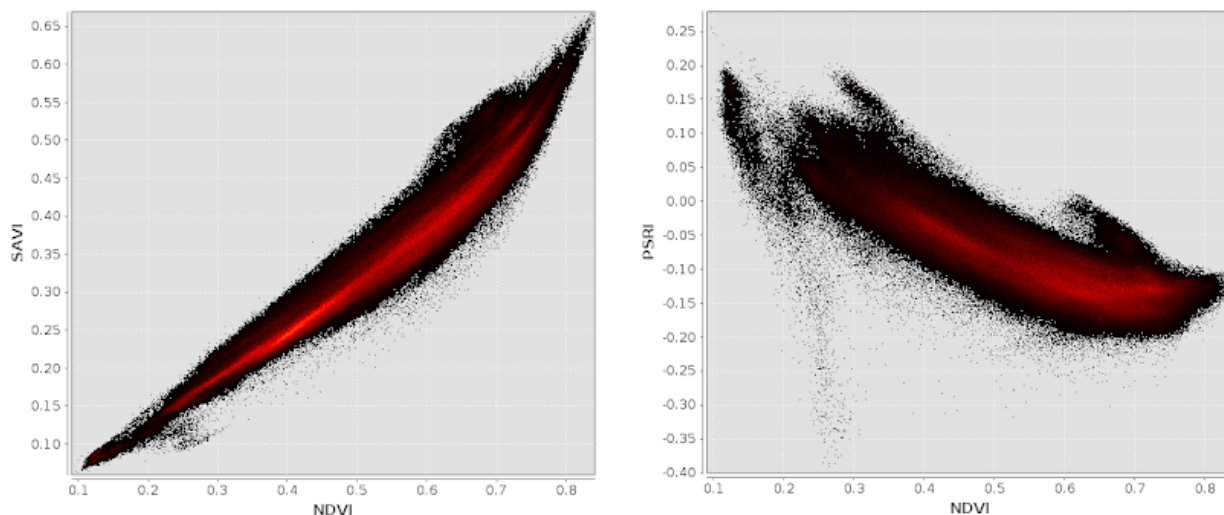
Käesoleva töö üheks eesmärgiks on leida indekseid, ehk Sentinel-2 erinevate spektrikanalite matemaatilisi kombinatsioone, mille närvivõrgu sisendina andmine närvivõrgule suurendaks klassifitseerimistäpsust.

Uute indeksite valimiseks arvutati välja potentsiaalsed indeksid SNAP-is, (ingl. k. Sentinel Application Platform) mis on ESA ametlik satelliidipiltide töötlemise tarkvara. Seejärel võrreldi potentsiaalse uue indeksi väärtusi NDVI indeksi väärtustega.



Joonis 3.5: Indeksite korrelatsiooni arvutamiseks kasutatud põllud on välja toodud roosaga. Kokku on põlde 1260 ning esindatud klasse 19.

Potentsiaalseid uusi indekseid võeti artiklitest [58], [59], [60], [61] ning indeksite andmebaasist [62]. Joonisel 3.6 on näha, kuidas korrelatsiooni alusel valiti indekseid. Eesmärk on valida indekseid, mis annaksid närvivõrgu sisendtunnuste komplekti uut infot.



Joonis 3.6: Vasakul on NDVI korrelatsioon indeksiga SAVI (ingl. k. Soil-Adjusted Vegetation Index) mida ei valitud. Paremal on näha NDVI korrelatsiooni indeksiga PSRI(ingl. k. Plant Senescence Reflectance Index), mis on selgelt vähem korreleeritud NDVI-ga, mistõttu antud indeks valiti närvivõrgu sisendtunnuste komplekti.

Tabelis 3.1 on esitatud kõik ribad ja indeksid, mida käesolevas töös närvivõrgu sisendiks võeti. Tabelist on Sentinel-2 ribadest puudu 10. riba 1375 nm juures, mis kannab infot ainult kiudpilvede kohta. Seda kasutatakse, et L1C tötluse tasemel pilvede peegeldust maha arvestada, mistõttu L2A pildidel pole 10. riba. Käesolevas töös kasutatakse ainult L2A töotlustaseme pilte. [63]

Sentinel-2 mõõdab kolme erineva lahutusvõimega:

- 10 meetrise lahutusvõimega on B2, B3, B4, B8
- 20 meetrise lahutusvõimega on B5, B6, B7, B8A, B11, B12
- 60 meetrise lahutusvõimega on B9, B10

Valituks osutunud indeksitest PSRI on mõeldud taimede kõdumise esile toomiseks, Misra Yellow Vegetation on disainitud reageerima kollast värvi taimedele ning Tasseled Cap indeksid moodustavad matemaatiliselt ortogonaalse kolmiku. [58] [65]

Indeksite mõju närvivõrgu lõppennustustele on võimalik arvuliselt hinnata, kuid närvivõrgu filtrite kaalud seda mõju ei kirjelda. Käesolevas töös oli pärast treenimist närvivõrgu esimeses kihis kõikide indeksite filtrite kaalude absoluutväärtuste summa vahemikus 27,26 kuni 29,09.

Tabel 3.1: Indeksite tabel. Sentinel-2 ribade valemid on kujul: keskne lainepikkus/riba laius. [64]

Kanali #	Kanali nimi	Valem või lainepikkuste vahemik
0	(TCW) Tasseled Cap Wetness	$0.1509 * B2 + 0.1973 * B3 + 0.3279 * B4 + 0.3406 * B8 - 0.7112 * B11 - 0.4572 * B12$
1	(TCV) Tasseled Cap Vegetation	$-0.2848 * B2 - 0.2435 * B3 - 0.5436 * B4 + 0.7243 * B8 + 0.0840 * B11 - 0.1800 * B12$
2	(TCB) Tasseled Cap Brightness	$0.3037 * B2 + 0.2793 * B3 + 0.4743 * B4 + 0.5585 * B8 + 0.5082 * B10 + 0.1863 * B12$
3	(MYV) Misra Yellow Vegetation	$0.723 * B3 - 0.597 * B4 + 0.206 * B6 - 0.278 * B9$
4	(PSRI) Plant Senescence Reflectance Index	$\frac{(B4 - B2)}{(B6)}$
5	(WRI) Water Ratio Index	$\frac{(B3 - B4)}{(B8 + B11)}$
6	(NDVIre) Red-edge Normalized Difference Vegetation Index	$\frac{(B8 - B6)}{(B8 + B6)}$
7	(NDVI) Normalized Difference Vegetation Index	$\frac{(B8 - B4)}{(B8 + B4)}$
8	(NPCI) Normalized Pigment Chlorophyll Index	$0.723 * B3 - 0.597 * B4 + 0.206 * B6 - 0.278 * B9$
9	(EVI) Enhanced Vegetation Index	$\frac{2.5 * (B8 - B4)}{B8 + 6 * B4 - 7.5 * B2 + 1}$
10	(B1) Sentinel-2 Band1	434,9nm/27nm
11	(B2) Sentinel-2 Band2	496,6nm/98nm
12	(B3) Sentinel-2 Band3	560,0nm/45nm
13	(B4) Sentinel-2 Band4	664,5nm/38nm
14	(B5) Sentinel-2 Band5	703,9nm/19nm
15	(B6) Sentinel-2 Band6	740,2nm/18nm
16	(B7) Sentinel-2 Band7	782,5nm/28nm
17	(B8) Sentinel-2 Band8	835,1nm/145nm
18	(B8A) Sentinel-2 Band8A	864,8nm/33nm
19	(B9) Sentinel-2 Band9	945,0nm/26nm
20	(B11) Sentinel-2 Band11	1613,7nm/143nm
21	(B12) Sentinel-2 Band12	2202,4nm/242nm

## Peatükk 4

# Implementatsioon

### 4.1 Töökeskkonna leidmine

Masinõppe algoritmi arendati esmalt isklikus sülearvutis jupyteri keskkonnas. Kiiresti sai selgeks, et 3D sidumnärvivõrk on väga ressursimahukas algoritm ning pelgalt sisendandmete hoidmiseks on vaja suurt kogust muutmälu (ingl. k. Random-Access Memory ehk RAM). Ajutine lahendus oli kasutada ESA arendus- ja tugikeskust (ingl. k. Research and User Support ehk RUS), mis võimaldas kasutada 60 GB RAM-i ning 300GB kõvaketta mälu virtuaalmasinat. [66]

RUS keskkond osutus siiski sobimatuks, sest seal puuduvad graafikakaardi (ingl. k. Graphics Processing Unit ehk GPU) kasutamise võimalused ning keskprotsessoriga (ingl. k. Central Processing Unit ehk CPU) ei oleks jõutud närvivõrku õigeks ajaks ära treenida. GPU võimaldab närvivõrku treenida 15 korda kiiremini kui keskprotsessor [67].

Valikus oli ka hiljuti valminud ESTHub, kuid seal puudus käsurea kasutamise võimalus, mistõttu tehti lõviosa tööst OÜ KappaZeta tööjaamas, kus on 64 GB RAM-i ning GeForce GTX 1070 8GB graafikakaart.

Töömahu kasvades muutus kriitiliseks ka programmeerimiskeskkond. Jupyter on suurepärase prototüüpimiseks, kuid mäluhalduse võimaluste puuduse tõttu mindi üle objektorienteeritud pythoni skriptide kirjutamisele ning nende jooksumisele käsurealt.

### 4.2 Keskkonna ülesseadmine

Erinevates arvutites töötamine oli võimalik tänu virtuaalsetele pythoni keskkondadele. Selleks, et käesoleva töö jaoks vajalike pythoni teekide installimisega ei tekiks konflikte arvuti teiste

kasutajate teekidega, tuli tekitada ülejäänud arvutist eraldatud virtuaalne pythoni keskkond, mis võimaldas alustada teekide paigaldamisega alati nullist. Töökeskkonna ülesseadmiseks pidi paigaldama konkreetse kombinatsiooni erinevate teekide versioonidest, õiges järjekorras ning õigetest allikatest.

Pythoni tekidest osutusid hädavajalikuks järgnevad:

- georaster ja rasterio olid vajalikud, et lugeda sisse ning töödelda koordinaatsüsteemidega seotud satelliitpiltide rastereid.
- pyshp oli vaja, et lugeda põldude ja uurimisala .shp vektorfaile, polügoone.
- shapely teisendas polügoon objektid õigesse vormi.
- scikit-image võimaldas kiiret madalatasemelist kahemõõtmelise pildi ümbrusteks tükeldamist.

Virtuaalse keskkonna ülesseadmisesest saab täpsemalt lugeda lisast A.2

## 4.3 Lõplik tulemus GitHubis

Käesoleva töö edukaks haldamiseks kasutati git versioonihaldustarkvara ning kogu töö käigus kirjutatud kood leidub githubi hoidlas [https://github.com/AkselAllas/Bachelor\\_thesis](https://github.com/AkselAllas/Bachelor_thesis) (edaspidi viidatud kui github hoidla).

Githubi hoidlas olevatele failidele viidatakse töös edaspidi lähtuvalt hoidla juurkausta './' asukohast, näiteks juurkaustas leidub kaust Scripts, mis sisaldab skripti ./Scripts/count\_class\_occurrences.py, mis loeb kokku kõikide klasside pikslite arvud. Täpsemalt githubi hoidlas oleva failide ning kaustade struktuuri kohta saab lugeda lisast A.4.

## 4.4 Andmete eeltöötlus

Enne närvivõrgu treenimist on vaja tagada lähteandmete vormingu ja masinõppe sisendi nõuete ühilduvus.

Vastavalt joonisele 2.2, tuleb närvivõrgu sisendiks anda iga piksli kohta tema 8x8 ümbrus ning seda iga 5 Sentinel-2 pildi 22 erineva sisendkihi jaoks. Tähendab, et närvivõrgu sisendtensor on mõõtetega (sisendite\_arv, 22, 8, 8, 5), kus sisendite arv on kahemõõtmelise pildi pikslite arv. Kuna käesoleva töö arhitektuur ennustab iga 8x8 sisendi kohta ühe piksli klassi, siis tuleb kahemõõtmelise pildi iga piksli jaoks anda 8x8 ümbrus närvivõrgu sisendiks. Töötamise mugavuseks tehti esmalt kahemõõtmeline pilt ühemõõtmeliseks 8x8 ümbruste

jadaks, mis anti närvivõrgule klassifitseerimiseks ning seejärel ennustuse kuvamiseks tehti taas ühedimensionaalsest jadast kahedimensionaalne pilt.

Arvestades, et Sentinel-2 ruudu mõõtmed on 10980x10980 pikslit, siis tuleks sisendtensori mõõtmeteks (120560400, 22, 8, 8, 5) ning selle tensori korraga muutmälus hoidmiseks kuluks 3394 GB RAM-i, arvestades, et iga väärtus on 32 bitiline ujukoma muutaja. Lisaks peab sisendtensoriga opereerima, mistõttu ei piisa kindlasti ainult 3394 GB-st.

Ainuke võimalus on lugeda andmeid treenimiseks sisse tükkhaaval. Selleks jaotatakse Sentinel-2 ruut väiksemateks osadeks.

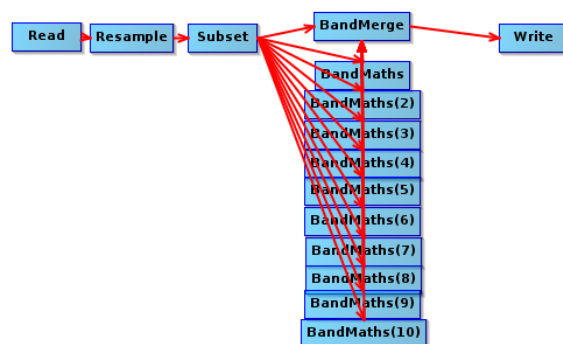
Tulenevalt arvutite ja programmeerimise põhinemisest kahendsüsteemil, on kasulik treenides lugeda sisse andmeid, mille mõõtmed on kahe astmed. Testimise käigus leiti, et maksimaalne kahe astme kordse mõõduga ruut, mida KappaZeta arvuti suudab korraga sisse lugeda on 512x512.

Sentinel-2 ruudust lõigati seega välja 10240x10240 pilt, mis mahutab 20x20 ruutu mõõtmetega 512x512. Selle tõttu võeti Sentinel-2 ruudult ära 720 pikslit mõlemast mõõdust. Joonisel 4.2 on toodud punasega piirkond, mis moodustab närvivõrgule sisendiks antava 10240x10240 piirkonna.

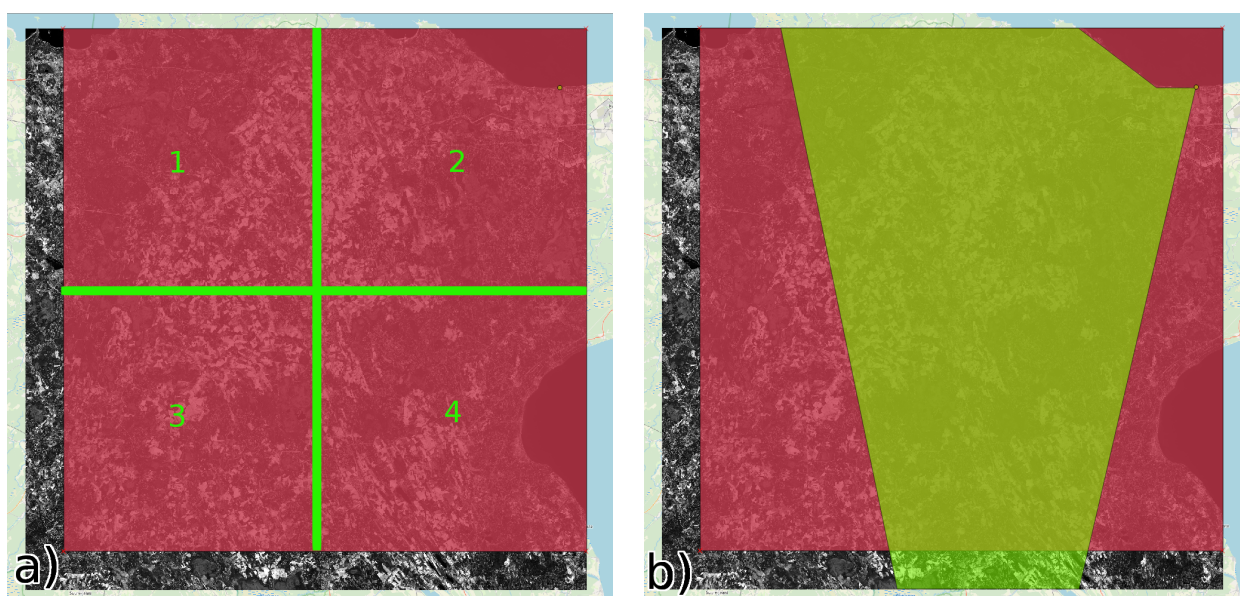
Selleks, et erineva lahutusvõimega sisendkihte ja indekseid oleks võimalik anda närvivõrgu sisendiks, tuleb 20 m ja 60 m lahutusvõimega indekseid teisendada võrdseks 10 m lahutusvõimega indeksitega. Selle teisenduse tegemiseks kasutatakse SNAP-i, mis võimaldab lahutusvõimete vahel teisendusi teha lähima naabri meetodil (ingl. k. 'upsample nearest'). SNAP-i kaasamine eeltöötlusahelasse on tähtis, sest see võimaldab tulevikus kasutada sealseid funktsioone, mis vähendavad pilvede osakaalu pildist.

Lisaks varasematele teisendustele arvutatakse SNAP-iga ka välja uued indeksid, kasutades SNAP-i töötlusgraafi, mida näeb joonisel 4.1. Tulenevalt uute indeksite lisamisest on salvestatav pilt suurem kui .tif failiformaadi limiidiks olev 4GB. Seega tuleb Sentinel-2 ruut jagada neljaks väiksemaks 5120x5120 ruuduks. Vt. joonis 4.2

Andmed märgendati närvivõrgu treenimiseks PRIA põllukultuuride vektorkihi abil. Täpsemalt saab tõeväärtuste sisendandmete töötlemisest lugeda lisast A.6 ning githubi hoidla ./Gpt\_scripts/ kaustast.



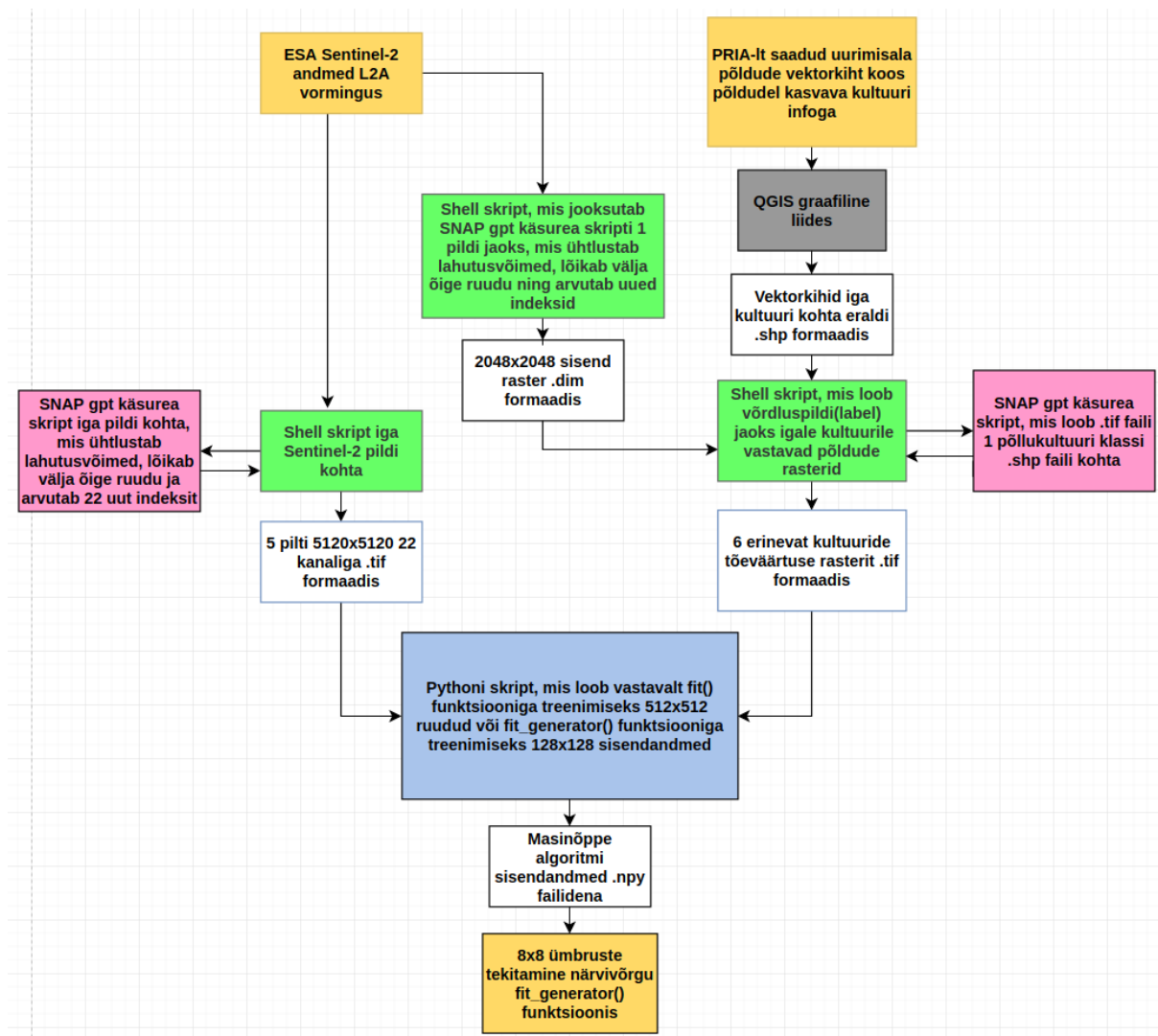
Joonis 4.1: SNAP-i töötlusgraaf, mis loeb Sentinel-2 pildi sisse, ühtlustab kõik lahutusvõimed 10m peale, lõikab välja õige 5120x5120 ruudu ning arvutab uued indeksid.



Joonis 4.2: a) Sentinel-2 pilt, kus on välja toodud punase alaga pildi osa, mis võetakse närvivõrgu sisendandmeteks. Failiformaadi .tif 4GB mahupiirangu tõttu tükeldatakse piirkond neljaks 5120x5120 pildiks. b) Sentinel-2 pilt, kus punase ala ning kollasega kujutatud optimaalse uurimisala ühisosa on lõplikuks närvivõrgu sisendiks.

Lõpuks valmis joonisel 4.3 kujutatud eeltöötlusahel, kus jookseb kaks erinevat haru. Vasakpoolne haru valmistab satelliidipiltidest sisendid närvivõrgule ning parempoolsele igale pikslile vastava tõeväärtuse.





Joonis 4.3: Eeltötlusahel lähteandmetest närvivõrgu sisendandmete töötlemiseks. Beežiga on toodud eeltötlusahela sisendid üleval ning väljund all. Rohelised on bash skriptid, roosad on SNAP graafid käsurea kujul, sinine on pythoni skript, hall on graafiline liides ning valged on failid erinevates vormingutes.

## 4.5 3D sidumnärvivõrgu näitest lähtumine

Projekti tehnilist arendamist alustati 3D CNN näite tööle saamisega, et veenduda alustehnoloogiad töötavad ootuspäraselt. Ainukeseks kättesaadavaks Tensorflow-il ja Keras-el põhinevaks 3D CNN implementatsiooniks osutus Eric Tsang ning Anuj Shah õpetus. [68], [69]

Arhitektuur muudeti analoogseks Shunping 2018 artikliga [50], muutes sisendparameetreid

väärtusi, lisades vastavad kihid ning kihtide vahele BatchNormalization operatsioonid. [70]

Suurt probleemi valmistas mudeli esialgne kalduvus ennustada kõik klassiks null, ehk et pildi peal polegi põllukultuure. Nullide klassifitseerimise kaotamiseks lisati enne viimast sõlme väljajätku operatsioon (ingl. k. dropout), mis treenides lülitab suvaliselt osade sõlmede aktivatsioonid nulli, et vähendada ülesobitus [71]. Lisaks arvutati igale klassile kulufunktsiooni kaal sõltuvalt klassi esinemissagedusest [72].

Loodi joonisel 4.4 välja toodud mudel.

Layer (type)	Output Shape	Param #
conv3d_1 (Conv3D)	(None, 32, 8, 8, 5)	19040
batch_normalization_1 (Batch Normalization)	(None, 32, 8, 8, 5)	128
activation_1 (Activation)	(None, 32, 8, 8, 5)	0
max_pooling3d_1 (MaxPooling3D)	(None, 32, 4, 4, 5)	0
conv3d_2 (Conv3D)	(None, 32, 4, 4, 5)	27680
batch_normalization_2 (Batch Normalization)	(None, 32, 4, 4, 5)	128
activation_2 (Activation)	(None, 32, 4, 4, 5)	0
max_pooling3d_2 (MaxPooling3D)	(None, 32, 2, 2, 5)	0
conv3d_3 (Conv3D)	(None, 64, 2, 2, 5)	55360
batch_normalization_3 (Batch Normalization)	(None, 64, 2, 2, 5)	256
activation_3 (Activation)	(None, 64, 2, 2, 5)	0
max_pooling3d_3 (MaxPooling3D)	(None, 64, 1, 1, 5)	0
flatten_1 (Flatten)	(None, 320)	0
dense_1 (Dense)	(None, 64)	20544
dropout_1 (Dropout)	(None, 64)	0
dense_2 (Dense)	(None, 28)	1820
activation_4 (Activation)	(None, 28)	0
Total params: 124,956		
Trainable params: 124,700		
Non-trainable params: 256		

Joonis 4.4: Käesolevas töös kasutatava 3D sidumärvivõrgu kihtide kokkuvõte

Treenitavate parameetrite arvu esimeses sidumkihis saab leida järgnevalt: Igal filtril on iga 3 ajalise pildi jaoks 3x3 kaalud. Igal kanalil, ehk indeksil 22-st, on eraldi 3x3x3 kaalud, ehk iga filter on nelja mõõtmeline, mõõtmetega (3,3,3,22). Kokku on filtreid 32 ning igal filtril on vabaliige. Kokku on esimeses sidumkihis seega  $3 \times 3 \times 3 \times 22 \times 32 + 32 = 19040$  treenitavat parameetrit. [37]

## Peatükk 5

## Tulemused

### 5.1 Põllukultuuride liigitamine 3D sidumnärvivõrguga

#### 5.1.1 Klassifitseerimise kogutäpsuse baastase

Peamine närvivõrgu täpsuse näitaja, mida kasutatakse on klassifitseerimise kogutäpsus, mis on defineeritud kui korrektselt klassifitseeritud pikslite arv jagatud kogu pikslite arvuga.

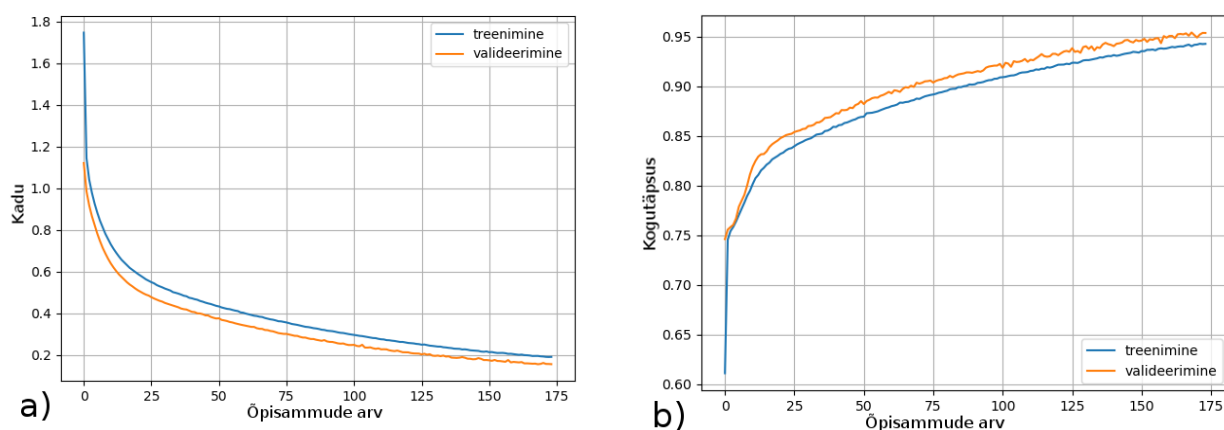
Kokku on käesoleva töö uurimisasalas 56,61% piksleid, kus ei leidu põllukultuuri. Kõige lihtsam närvivõrk ennustaks, et uurimisasalas pole mitte ühtegi põldu. See oleks väga kasutu närvivõrk, kuid ikkagi oleks saavutatud 56,61%-line kogutäpsus. Järelikult kui treenitud närvivõrk suudab põlde klassifitseerida suurema kogutäpsusega kui 56,61 %, siis saab väita, et närvivõrk klassifitseerib baastasemest täpsemini.

#### 5.1.2 Esialgne närvivõrgu treenimine

Baastasemest parema täpsuse saavutamiseks otsiti andmekogust 512x512 ruut, kus esineb kõige rohkem erinevate klasside põlde. Selleks kasutati githubi hoidlas olevat skripti `./Big_tile_data/get_tile_unique_classes.py` ning valituks osutus ruut, kus on esindatud 21 klassi 28-st.

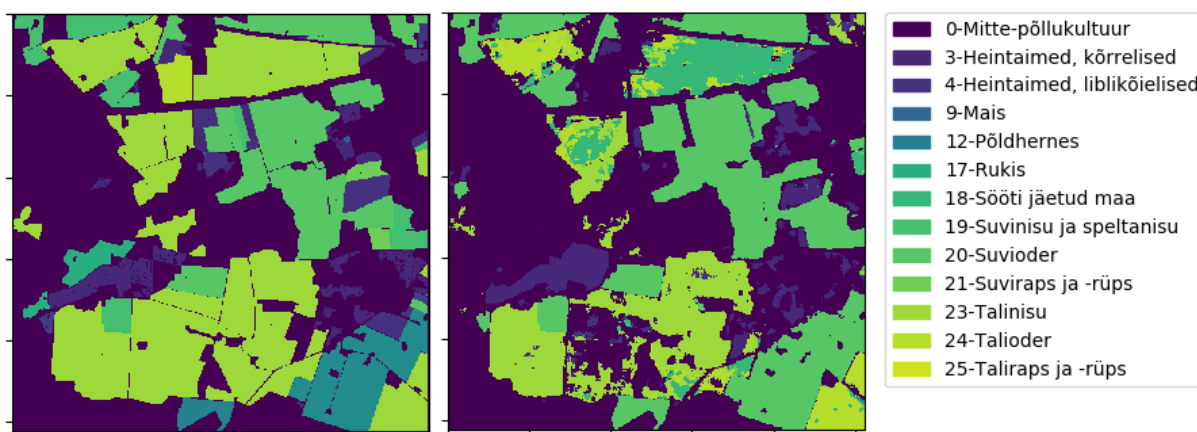
Närvivõrku treeniti stohhastilise gradientlaskumise optimeerijaga (ingl. k. Stochastic Gradient Descent ehk SGD), kasutades õpisammu väärtusena 0,001. Sisendandmeteks oli ainult üks 512x512 ruut ning üks õpisamm tehti iga 500 piksli tagant, artikli [50] eeskujul. Väljajäta tehnikat ei kasutata. Samuti [50] eeskujul võeti tavapärasest suurem valideerimisandmestik, ehk 50% 512x512 ruudu pikslitest kasutati valideerimiseks, ehk nende pikslite peal ei treenita, vaid

pärast iga õpissammu arvutatakse valideerimisandmestiku peal kao väärtus. Ülejäänud 50% pikslitest kuulus treeningandmestikku. Ülesobituse vältimiseks kasutati varajast peatamist, mis lõpetas treenimise juhul kui valideerimise kao väärtus suurenes viiel järjestikkusel sammul. Nimetatud treenimise kao ja kogutäpsuse kujunemist näeb joonistel 5.1-a ja 5.1-b.



Joonis 5.1: a) Treenides ühel 512x512 ruudul, väheneb kaovärritus väga sujuvalt ning kiiresti. b) Ühel ruudul treenimisega sobitatakse närvivõrk treenitavale ruudule üle, saavutades sellel ruudul küll suure kogutäpsuse, kuid saadud mudel ei üldistu närvivõrgu jaoks tundmatutele ruutudele.

Pärast ühel ruudul treenimist suudab närvivõrk oma ülesobitatud andmekogu ennustada 94,85% kogutäpsusega ning närvivõrgu jaoks tundmatut ruutu 65,59%-lise kogutäpsusega, nagu näha joonisel 5.2.



Joonis 5.2: Vasakul joonisel on närvivõrgu jaoks tundmatu ruutu tõeväärtused ning paremal närvivõrgu tehtud ennustused. Kogutäpsus 65,6%.

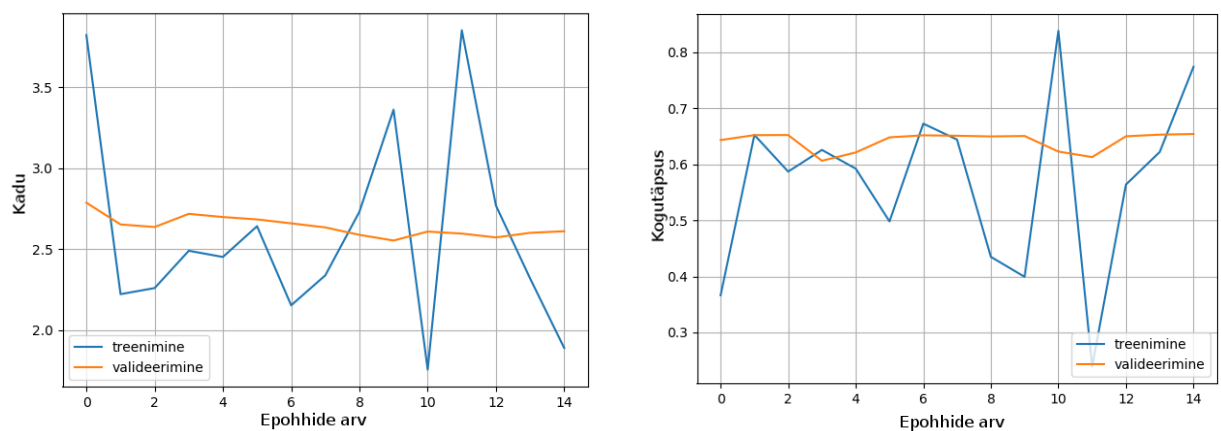
Järgnevalt võrreldakse kahte närvivõrgu treenimise lähenemist.

### 5.1.3 Närvivõrgu treenimine tehes ühe sammu kogu andmekogu nägemise peale

Kogu andmestiku sisselugemiseks kasutati Kerase sisseehitatud funktsiooni `fit_generator()` koos spetsiaalselt käesolevas töös kasutatava närvivõrgu jaoks modifitseeritud generaatoriga, mille leiab github hoidlast failis `./data_generator.py`.

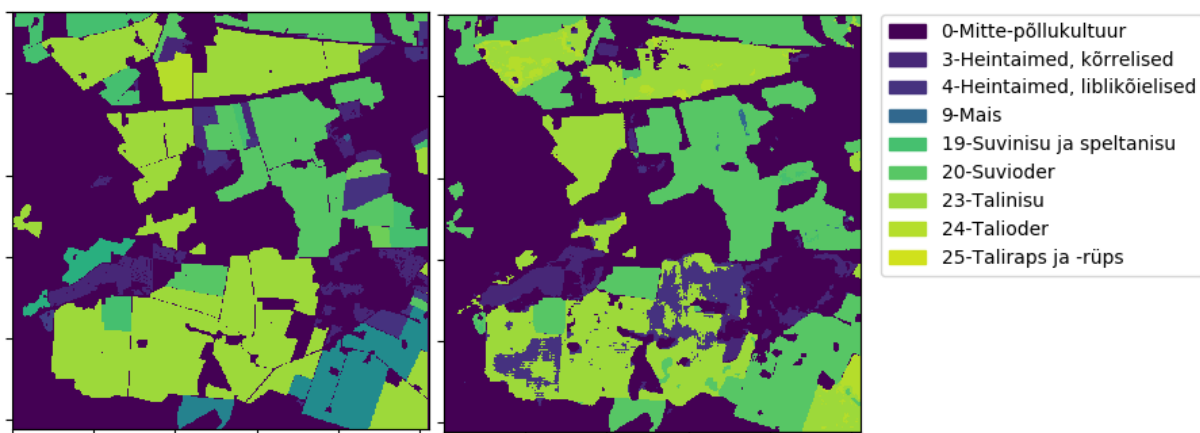
Väljatoodud generaator loeb sisse sisendfailid ning teeb andmetega viimase eeltöötluste, nimelt kasutab pythoni teegi `skimage` funktsiooni `view_as_windows`, et tekitada igale pikslile 8x8 ümbrus. Iseloodud generaatori kasutamine võimaldab kasutada mitut lõime, kus üks trenib mudelit GPU-l ning teine valmistab ette järgmist sisendandmete hulka. Korraga valiti sisendandmeteks 128x128 ruudud, sest suuremaid ruute ei suudetud `fit_generator` puhul mälus hoida. Mälu kulub vastavalt avaldisele:  $(\text{sisendtensori maht}) \times (\text{lõimede arv}) \times (\text{maksimaalne järjekorra arv}) \times 2$ . Eelmise avaldise 2 tuleb nii treenimise kui ka valideerimise vajadusest.

Mudeli treenimist jätkati joonises 5.1 treenitud mudelist, muutes õpisammu väärtuse 0.0001 peale ning jättes muud hüperparameetrid samaks. Joonisel 5.3 on näha kogu andmestiku peal treenimise tulemus.



Joonis 5.3: Kogu andmestiku peal treenimise tulemus. Nii vasakul oleva kadu kujutava kui ka paremal oleva kogutäpsust kujutava graafiku puhul on funktsioonid ebastabiilsed, viidates probleemile treenimisstrateegias.

Pärast kogu andmestiku peal treenimist suutis närvivõrk tundmatut ruutu ennustada 73,50%-lise kogutäpsusega, nagu näha joonisel 5.4. Täieliku andmekogu peal treenitud närvivõrk sai kümne tundmatu 512x512 ruudu keskmiseks täpsuseks 76,57%.

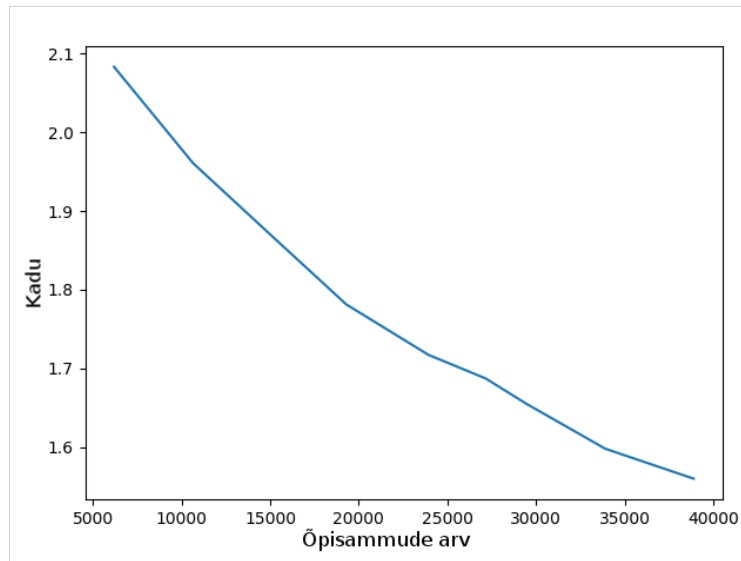


Joonis 5.4: Vasakul joonisel on närvivõrgu jaoks tundmatu ruudu tõeväärtused ning paremal närvivõrgu tehtud ennustused. Kogutäpsus 73,50%.

#### 5.1.4 Närvivõrgu treenimine tehes ühe sammu 500 piksli nägemise peale

Parim kogutäpsus saavutati strateegiaga, kus treeniti terve andmestikuga, tehes õpisamme iga 500 piksli tagant. Treenimisel oli õpisammu väärtuseks 0,001 ning muud hüperparameetrid olid varasemate strateegiatega samad. Täiendavaid regulariseerivaid meetmeid ei kasutatud.

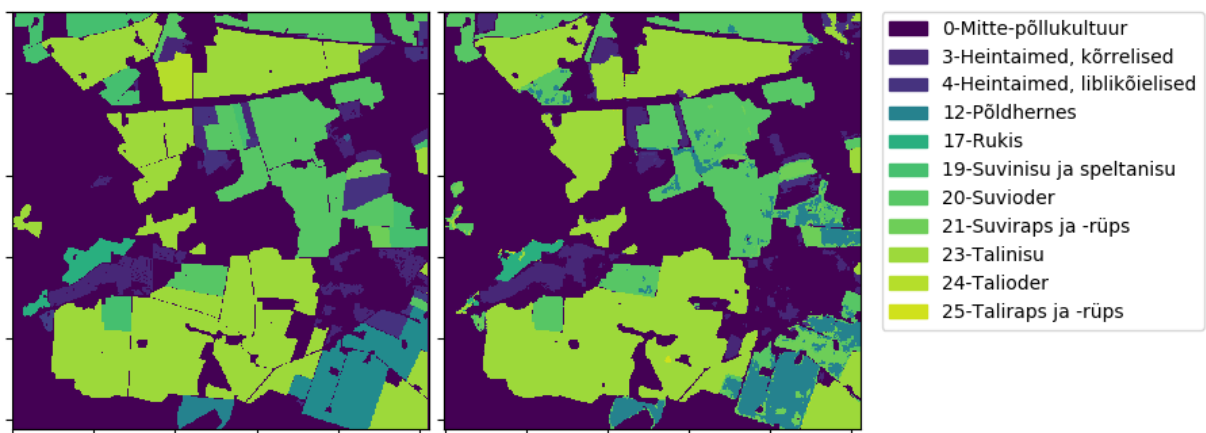
Kuna ühe epohhi treenimiseks kulub 51 tundi, siis ei olnud võimalik automaatselt logida treenimise ajal valideerimiskadu ja -täpsust. Selle asemel märgiti käsitsi 10 kao väärtust, mida näeb joonisel 5.5.



Joonis 5.5: Tehes õpisammu iga 500 piksli järel, on võimalik saavutada madalam kaoväärtus võrreldes seadega, kus tehti üks õpisamm terve andmestiku peal.

Pärast kogu andmestiku läbi töötamist, treeniti veelkord kolmandikul andmestikust ning võrreldi kogutäpsusi samal kümnel tundmatul 512x512 ruudul, mida kasutati ka terve andmestiku peale õpisamme teinud mudeli hindamiseks. Vana mudeli keskmine täpsus kümnel ruudul oli 81,41%, samas kui uue mudeli täpsus oli 81,30%. Pärast seda treeniti veel kolmandiku andmestikuga, kuid kasutades kõikidel sidumkihtidel 11 regulariseerimist väärtusega 0.01. Kõige uuema mudeli keskmine täpsus kümnel ruudul oli 82,92%.

Võrreldes varasemate treenimisstrateegiatega, saavutab uus mudel võrdlusruudul 81,65%-lise kogutäpsuse, mida on näha joonisel 5.6.



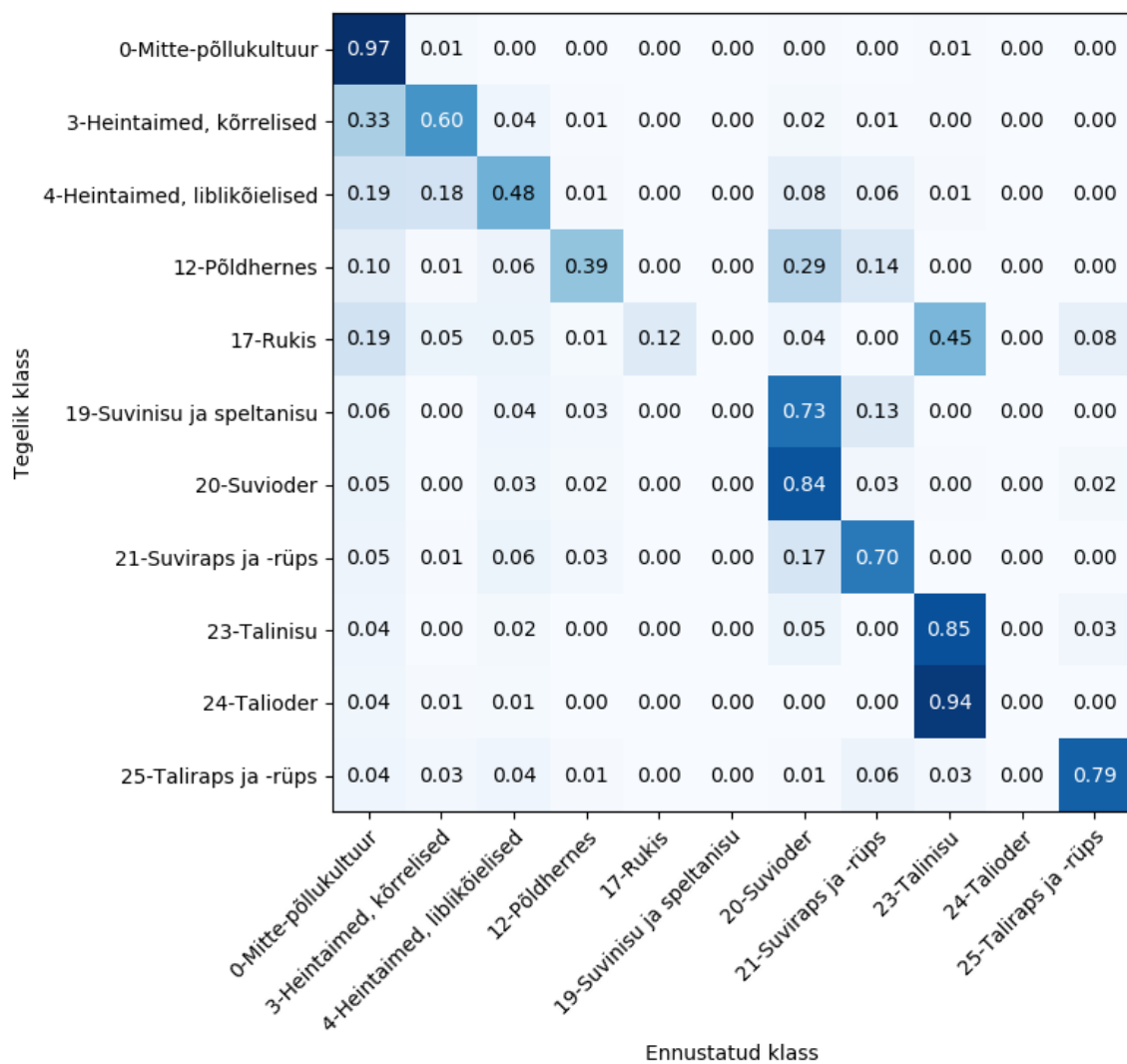
Joonis 5.6: Põllukultuuride ennustuste kaart tehtud närvivõrguga, mis sammus kogu andmestiku peal 500-pikslise sammuga. Vasakul joonisel on närvivõrgu jaoks varem nägemata ruudu tõeväärtused ning paremal närvivõrgu tehtud ennustused. Kogutäpsus 81,65%.

Sammudes 500 piksli tagant, suudab närvivõrk taas õppida rohkem klasse ennustama. Ka põldude piire suudab kõige uuem närvivõrk väga edukalt ära tunda.

Kõige parema täpsusega närvivõrgu ennustust kõige suurema põldude tihedusega 1024x1024 piirkonnale saab näha lisas A.8.

Närvivõrkude võrdlemisel ei saa piirduda vaid kogutäpsusega, vaid arvestama peab ka valede positiividega (ingl. k. false positive). Kõige ülevaatlikum viis mudeli täpsuse hindamiseks on veamaatriksi joonestamine. Joonisel 5.7 on näha võrdluseks kasutatud kümne 512x512 ruudu andmetega tehtud normeeritud veamaatriksit. Veamaatriksi ühe rea moodustavad ühe klassi pikslid, mida mudel on ennustanud erinevatesse klassidesse. Normeeritud veamaatriksis on ühe rea elementide summa 1. Mida diagonaalsem on veamaatriks, seda täpsem on närvivõrk. Ideaalse närvivõrgu veamaatriks oleks ortogonaalne.





Joonis 5.7: Parima kogutäpsusega närvivõrgu veamaatriks kümne 512x512 ruudu sisendandmetega. Välja on toodud ainult klassid, mida närvivõrk oskas ennustada.

Veamaatriksist on näha, et mitte-põllukultuuri klassi täpsus on 97%, mis tähendab et põldude piirid on väga hästi paigas. Lisaks on näha, et närvivõrk ajab omavahel sassi teravilju nagu talinisu, talioder ja rukis. Veel on närvivõrgul raske eristada suviotra ja suvinisu.

Normeerimata veamaatriksit saab vaadata lisa A.7 ning sealt on näha, et mida rohkem on klassis piksleid, seda suurema täpsusega sinna klassi liigitatakse. See tähendab, et käesolevat mudelit saab kindlasti paremaks treenida, leides optimaalsed hüperparameetrid ja regulariseerimise astme. Lisaks võib see viidata vajadusele anda väiksemate klasside kohta rohkem sisendandmeid.

Lõpliku mudeli ennustused tundmatutel ruutudel on toodud lisa A.8.

## 5.2 Arutelu

3D sidumnärvivõrk suutis väga hästi tuvastada põldude piire. Seda mudeli oskust saaks rakendada, et tuvastada Eestis põlde, mis pole veel PRIA andmebaasides.

Järgmise uurimise sammuna tuleb leida viisid klassifitseerimistäpsuse maksimeerimiseks ning seejärel klasside haaval analüüsida tulemusi võrreldes näiteks Peter Gelsbo magistritöö [54] ning teiste sarnase detailsusastmega töödega.

Kui klassifitseerida põldude kaupa, andes põllule klassi, mille piksleid on põllu vektori sees kõige rohkem, suureneks klassifitseerimistäpsus, samamoodi nagu oli Peter Gelsbo töös [54].

Kindlasti suurendaks täpsust geomeetiline sisendandmete rikastamine. Selle asemel, et sisendandmeid pöörata, võiks proovida artiklis [73] kasutatud meetodit pöörata filtreid.

Kogutäpsuse suurendamiseks tasub proovida närvivõrgu treenimist erinevate üldistavate meetoditega nagu L1, L2 regularisatsioon, kaalude väljavõtt (ingl. k. dropout) ja kaalude kahandamine.

Täpsust aitaks suurendada ka Sentinel-2 piltidele pilvemaskide rakendamine, interpoleerides pilvede poolt kaetud pikslite väärtusi vastavate pikslite Sentinel-2 piltide aegridade abil.

Kategoorilise ristentroopia kaofunktsiooni asemel võiks proovida ka üldistatud täringu ühisosa kaofunktsiooni (ingl. k. Generalised Dice loss function) [74]. Lisaks tasuks uurida, kui suurel määral mõjutab päikese tekitatud varjude esinemine põldudel Sentinel-2 mõõtetulemusi ning kui lisada närvivõrgu sisendina piksli kaugus Sentinel-2 detektorist, kas klassifitseerimistäpsus läheks paremaks.

Tähtsaimate sisendtunnuste tuvastamiseks peaks võib proovida artiklis [75] välja käidud 'eemalda ja õpeta ümber' meetodit (ingl. k. RemOve And Retrain ehk ROAR).

Edasiseks võiks arendada lisaks Sentinel-2 mudelile ka Sentinel-1 piltidel töötava 3D sidumnärvivõrgu. Operatiivseks kasutuseks on Sentinel-1 pilte kindlasti vaja, sest Sentinel-2 sisendiks sobivad vaid pilvevabad pildid ning Eesti tingimustes on sõltuvalt ilmast ja aastaajast pilvituid Sentinel-2 pilte vahel väga hõredalt. Lisaks Sentinel-2 mõõtmistele saab mudeli sisendina kasutada ka meteoandmeid ja mullakaarte.

# Kokkuvõte

Käesolevat uurimust alustati erinevate kaasaegsete masinõppe arhitektuuride võrdlemisega. Uurimiseks valiti kõige suurema potentsiaaliga süvaõppe arhitektuur.

Selleks et tulevikus oleks võimalik saadud tulemusi objektiivselt hinnata teistsuguste süvaõppe mudelitega, defineeriti optimaalne uurimisala lähtudes Sentinel-1 ja Sentinel-2 satelliidipiltide kaadripiiridest ja Eesti põllumajandusmaa tihedusest. Närvivõrgu klassifitseerimistäpsuse suurendamiseks valiti täiendavateks närvivõrgu sisendtunnusteks erinevaid vegetatsiooniindekseid, võrreldes indekseid omavahelisi sõltuvusi.

Käesolevas töös arendati välja Sentinel-2 andmete eeltöötlusahel ning kirjutati närvivõrgu treenimiseks suure taaskasutatavusega objektorienteeritud tarkvarakomponendid. Lisaks valmis suuremahulise andmekogu treenimise vajadustele vastav andmete generaator.

Projektis kasutati ainult Sentinel-2 pilte, mille pilvisus oli alla 5%. Andmekogu täiendamiseks võib tulevikus valida ka pilvisemaid pilte, interpoleerides aegride põhjal pilviste aladele vastavaid piksliväärtusi.

3D sidumnärvivõrgu rakendamine põllukultuuride pikselhaaval klassifitseerimiseks osutus paljulubavaks. Kasutades kõigest viit Sentinel-2 pilti maikuust, oli võimalik saavutada 82,92%-line klassifitseerimise täpsus tundmatul andmekogul. Olenemata 28 erinevast liigitamise klassist, suutis treenitud närvivõrk suurepäraselt tuvastada põldude piire, klassifitseerides mitte-põllukultuuri klassi 97%-lise täpsusega.

Klassifitseerimises saavutasid suurema täpsuse kõige suurema sisendite arvuga klassid. Järelikult on potentsiaali suurendada liigitamise täpsust, suurendades õpetusandmekogu, leides optimaalsed hüperparameetrid ning rakendades täiendavaid regulariseerivaid meetmeid.

Uurimuse päriseluliseks rakendamiseks on tarvis kaasata närvivõrgu andmeallikateks Sentinel-1 satelliite, kuna Sentinel-2 pilves ilmaga andmeid koguda ei saa. Lisaks tasub võtta mudeli sisendiks kasutusele meteoandmed ja mullakaardid.

# Tänuavaldused

Tahan väga tänada oma juhendajaid Kaupo Voormansikut ja Indrek Sünterit noorema põlvkonna koolitamise ning inspireerimise eest!

Suur aitäh kogu KappaZeta OÜ meeskonnale, kes on võimaldanud mul olla osa väga inspireerivast keskkonnast.

Täna Põllumajanduse Registrate ja Informatsiooni Ametit alti koostöö ning lähteandmete jagamise eest.

Lõpuks tahan tänada oma abikaasat, Katri-Helena Allast, kes tõesti aitab mul olla parem inimene ning on mind ohtralt toetanud sel teekonnal!

Aksel Allas

# Kirjandus

- [1] Jonathan A. Foley, Ruth DeFries, Gregory P. Asner, Carol Barford, Gordon Bonan, Stephen R. Carpenter, F.Štuart Chapin, Michael T. Coe, Gretchen C. Daily, Holly K. Gibbs, Joseph H. Helkowski, Tracey Holloway, Erica A. Howard, Christopher J. Kucharik, Chad Monfreda, Jonathan A. Patz, I. Colin Prentice, Navin Ramankutty, and Peter K. Snyder. Global consequences of land use. *Science*, 309(5734):570–574, 2005.
- [2] David Tilman, Christian Balzer, Jason Hill, and Belinda L. Befort. Global food demand and the sustainable intensification of agriculture. *Proceedings of the National Academy of Sciences*, 108(50):20260–20264, 2011.
- [3] Deepak Ray, Nathaniel Mueller, Paul West, and Jonathan A Foley. Yield trends are insufficient to double global crop production by 2050. *PloS one*, 8:e66428, 06 2013.
- [4] P. A. Matson, W. J. Parton, A. G. Power, and M. J. Swift. Agricultural intensification and ecosystem properties. *Science*, 277(5325):504–509, 1997.
- [5] Elena Bennett, Stephen Carpenter, and NINA F. CARACO. Human impact on erodable phosphorus and eutrophication: A global perspective. *BioScience*, 51:227–234, 01 2009.
- [6] Stuart Pimm and Peter Raven. Biodiversity - extinction by numbers. *Nature*, 403:843–5, 03 2000.
- [7] Stanley Wood, Carol Rosen, Hyacinth Billings, Maggie Powell, Kate Sebastian, Sara Scherr, Niels Batjes, Andrew Farrow, Jean Marc Faur, Gnther Fischer Iiasa, Gerhard Heilig Iiasa, Julio Henao Ifdc, Robert Hijmans, F Nachtergaele, Peter Oram Ifpri, and Manuel Winograd. *Pilot Analysis of Global Ecosystems: Agroecosystems*. 03 2001.
- [8] Claire Kremen, Neal M. Williams, and Robbin W. Thorp. Crop pollination from native bees at risk from agricultural intensification. *Proceedings of the National Academy of Sciences*, 99(26):16812–16816, 2002.
- [9] John Rogan and DongMei Chen. Remote sensing technology for mapping and monitoring land-cover and land-use change. *Progress in Planning*, 61(4):301–325, may 2004.

- [10] Xaviera A. López-Cortés, Fabiane M. Nachtigall, Verónica R. Olate, Macarena Araya, Sandra Oyanedel, Veronica Diaz, Eva Jakob, Mauricio Ríos-Momberg, and Leonardo Š. Santos. Fast detection of pathogens in salmon farming industry. *Aquaculture*, 470:17 – 24, 2017.
- [11] Rosaria Fragni, Antonio Trifirò, Ada Nucci, Andrea Seno, Alessia Allodi, and Matteo Di Rocco. Italian tomato-based products authentication by multi-element approach: A mineral elements database to distinguish the domestic provenance. *Food Control*, 93:211 – 218, 2018.
- [12] Camila Maione and Rommel Barbosa. Recent applications of multivariate data analysis methods in the authentication of rice and the most analyzed parameters: A review. *Critical Reviews in Food Science and Nutrition*, pages 00–00, 01 2018.
- [13] Kuai Fang, Chaopeng Shen, Daniel Kifer, and Xiao Yang. Prolongation of smap to spatio-temporally seamless coverage of continental us using a deep learning neural network. *Geophysical Research Letters*, 07 2017.
- [14] Flavio Barboza, Herbert Kimura, and Edward Altman. Machine learning models and bankruptcy prediction. *Expert Systems with Applications*, 83:405 – 417, 2017.
- [15] Yang Zhao, Jianping Li, and Lean Yu. A deep learning ensemble approach for crude oil price forecasting. *Energy Economics*, 66, 06 2017.
- [16] Marko Bohanec, Mirjana Borstnar, and Marko Robnik-Sikonja. Explaining machine learning models in sales predictions. *Expert Systems with Applications*, 71:416–428, 04 2017.
- [17] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large scale image recognition. *arXiv 1409.1556*, september 2014.
- [18] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. 7, 12 2015.
- [19] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Region-based convolutional networks for accurate object detection and segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38:1–1, 12 2015.
- [20] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. pages 779–788, 06 2016.
- [21] E. Shelhamer, J. Long, and T. Darrell. Fully convolutional networks for semantic segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(4):640–651, April 2017.

- [22] H. Noh, S. Hong, and B. Han. Learning deconvolution network for semantic segmentation. pages 1520–1528, Dec 2015.
- [23] Chee Seng Chan John E. Ball, Derek T. Anderson. Comprehensive survey of deep learning in remote sensing: theories, tools, and challenges for the community. *Journal of Applied Remote Sensing*, 11(4):1 – 54 – 54, 2017.
- [24] Konstantinos Liakos, Patrizia Busato, Dimitrios Moshou, Simon Pearson, and Dionysis D. Bochtis. Machine learning in agriculture: A review. In *Sensors*, 2018.
- [25] Euroopa liidu põllumajandustoetuste poliitika. [https://ec.europa.eu/info/food-farming-fisheries/key-policies/common-agricultural-policy/cap-glance\\_en](https://ec.europa.eu/info/food-farming-fisheries/key-policies/common-agricultural-policy/cap-glance_en). Vaadatud: 2019-03-28.
- [26] Esa sentinel-1 dokumentatsioon. <https://sentinel.esa.int/web/sentinel/user-guides/sentinel-1-sar>. Vaadatud: 2019-03-28.
- [27] Esa sentinel-2 dokumentatsioon. <https://sentinel.esa.int/web/sentinel/user-guides/sentinel-2-msi>. Vaadatud: 2019-03-28.
- [28] Luis A. Anchordoqui. Symmetries and the fundamental forces of Nature II. *arXiv e-prints*, Dec 2015.
- [29] Valter Kiisk. Spektroskoopia. loengukonspekt kursusele Itfy.01.014. <http://kodu.ut.ee/~kiisk/spec.pdf>, 2018. Vaadatud: 2019-05-26.
- [30] Euroopa kosmoseagentuur. Sentinel-2 väljundtoodete spetsifikatsioon. <https://sentinel.esa.int/web/sentinel/technical-guides/sentinel-2-msi/level-1b/product-formatting>, 2014. Vaadatud: 2019-05-26.
- [31] John B. Adams and Alan R. Gillespie. *Remote Sensing of Landscapes with Spectral Images: A Physical Modeling Approach*. Cambridge University Press, 2006.
- [32] Wikimedia Commons. Lamberti koosinuse seadus. [https://upload.wikimedia.org/wikipedia/commons/thumb/2/25/Lambert\\_Cosine\\_Law\\_1.svg/1024px-Lambert\\_Cosine\\_Law\\_1.svg.png](https://upload.wikimedia.org/wikipedia/commons/thumb/2/25/Lambert_Cosine_Law_1.svg/1024px-Lambert_Cosine_Law_1.svg.png), 2008. Vaadatud: 2019-05-26.
- [33] Sentinel-2 tehniline käsiraamat. [https://sentinel.esa.int/documents/247904/685211/Sentinel-2\\_User\\_Handbook](https://sentinel.esa.int/documents/247904/685211/Sentinel-2_User_Handbook), 2014. Vaadatud: 2019-05-26.
- [34] GISgeography. Spectral signature cheatsheet – spectral bands in remote sensing. <https://gisgeography.com/spectral-signature/>, Feb 2018. Vaadatud: 2019-05-26.
- [35] Vikipeedia ndvi lehekülg. [https://en.wikipedia.org/wiki/Normalized\\_difference\\_vegetation\\_index](https://en.wikipedia.org/wiki/Normalized_difference_vegetation_index), 2018. Vaadatud: 2019-05-26.

- [36] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. Deep learning. 2016. <http://www.deeplearningbook.org>.
- [37] Coursera masinõppe kursus. <https://www.coursera.org/learn/neural-networks-deep-learning/>, 2019. Vaadatud: 2019-04-01.
- [38] Aktiveerimisfunktsioonid närvivõrkudes. <https://towardsdatascience.com/activation-functions-neural-networks-1cbd9f8d91d6>, 2019. Vaadatud: 2019-04-01.
- [39] X. X. Zhu, D. Tuia, L. Mou, G. Xia, L. Zhang, F. Xu, and F. Fraundorfer. Deep learning in remote sensing: A comprehensive review and list of resources. *IEEE Geoscience and Remote Sensing Magazine*, 5(4):8–36, Dec 2017.
- [40] N. Kussul, S. Skakun, A. Shelestov, and O. Kussul. The use of satellite sar imagery to crop classification in ukraine within jecam project. pages 1497–1500, July 2014.
- [41] G. Cheng, J. Han, and X. Lu. Remote sensing image scene classification: Benchmark and state of the art. *Proceedings of the IEEE*, 105(10):1865–1883, Oct 2017.
- [42] Mit loeng kaasaegsetest arengutest masinõppes. <https://www.youtube.com/watch?v=53YvP6gdD7U>, 2019. Vaadatud: 2019-03-28.
- [43] Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. *CoRR*, abs/1603.02754, 2016.
- [44] Semantilise segmenteerimise võistlus pascal voc andmekogul. <https://paperswithcode.com/sota/semantic-segmentation-pascal-voc-2012>, 2018. Vaadatud: 2019-04-01.
- [45] Semantilise segmenteerimise võistlus cityscapes andmekogul. <https://paperswithcode.com/sota/semantic-segmentation-cityscapes>, 2018. Vaadatud: 2019-04-01.
- [46] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. *CoRR*, abs/1706.05587, 2017.
- [47] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. *CoRR*, abs/1802.02611, 2018.
- [48] Andres Camilo Rodriguez. Euroopa kosmoseagentuuri konverents sügavõppest kaugseires. esitlus satelliitpiltide pikselhaaval klassifitseerimise kohta. <https://lps19.esa.int/NikalWebsitePortal/living-planet-symposium-2019/lps19/Agenda/AgendaItemDetail?id=08b3009e-10f8-481a-8863-ccb45436b2da>, 2019. Vaadatud: 2019-05-16.



- [49] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. *CoRR*, abs/1505.04597, 2015.
- [50] Shunping Ji, Zhang Chi, Anjian Xu, Yun Shi, and Yulin Duan. 3d convolutional neural networks for crop classification with multi-temporal remote sensing images. *Remote Sensing*, 10:75, 01 2018.
- [51] Bharath Hariharan, Pablo Andrés Arbeláez, Ross B. Girshick, and Jitendra Malik. Hypercolumns for object segmentation and fine-grained localization. *CoRR*, abs/1411.5752, 2014.
- [52] N. Kussul, M. Lavreniuk, S. Skakun, and A. Shelestov. Deep learning classification of land cover and crop types using remote sensing data. *IEEE Geoscience and Remote Sensing Letters*, 14(5):778–782, May 2017.
- [53] Yuntian Chen, Haibin Chang, Meng Jin, and Dongxiao Zhang. Ensemble neural networks (enn): A gradient-free stochastic method. *Neural Networks*, 110, 12 2018.
- [54] Peter Gelsbo. Crop classification using deep learning. *Technical University of Denmark Master’s thesis*, 2019.
- [55] Runhani. 3d cnn võrdlus 2d ja 1d cnn-iga. <https://stackoverflow.com/questions/42883547/intuitive-understanding-of-1d-2d-and-3d-convolutions-in-convolutional-neural-n>, 2017. Vaadatud: 2019-05-16.
- [56] Euroopa kosmoseagentuur. Euroopa kosmoseagentuuri copernicuse programmi andmete avalik veebileht. <https://scihub.copernicus.eu/dhus/#/home>, 2014. Vaadatud: 2019-05-19.
- [57] Sentinel hub pilvevabade piltide leidmise tööriist. <https://apps.sentinel-hub.com/sentinel-playground>, 2019. Vaadatud: 2019-04-04.
- [58] A. Bannari, D. Morin, F. Bonn, and A. R. Huete. A review of vegetation indices. *Remote Sensing Reviews*, 13(1-2):95–120, 1995.
- [59] Caren C Dymond, David J Mladenoff, and Volker C Radeloff. Phenological differences in tasseled cap indices improve deciduous forest classification. *Remote Sensing of Environment*, 80(3):460 – 472, 2002.
- [60] Stavros Stagakis, Nikos Markos, Olga Sykioti, and Aris Kyparissis. Monitoring canopy biophysical and biochemical parameters in ecosystem scale using satellite hyperspectral imagery: An application on a phlomis fruticosa mediterranean ecosystem using multiangular chris/proba observations. *Remote Sensing of Environment*, 114(5):977 – 994, 2010.

- [61] Martín F. Garbulsky, Josep Peñuelas, Romà Ogaya, and Iolanda Filella. Leaf and stand-level carbon uptake of a mediterranean forest estimated using the satellite-derived reflectance indices evi and pri. *International Journal of Remote Sensing*, 34(4):1282–1296, 2013.
- [62] Katharina Brüser Verena Henrich. Index database. <https://www.indexdatabase.de/db/i-single.php?id=0>, 2011. Vaadatud: 2019-05-19.
- [63] Euroopa kosmoseagentuur. Sentinel-2 andmetöötlemissprogrammi sen2cor manuaal. <http://step.esa.int/main/third-party-plugins-2/sen2cor/>, 2018. Vaadatud: 2019-05-19.
- [64] Euroopa kosmoseagentuur. Sentinel-2 lahutusvõime andmeleht. <https://sentinel.esa.int/web/sentinel/missions/sentinel-2/instrument-payload/resolution-and-swath>, 2014. Vaadatud: 2019-05-19.
- [65] Eric P Crist. A tm tasseled cap equivalent transformation for reflectance factor data. *Remote Sensing of Environment*, 17(3):301 – 306, 1985.
- [66] Euroopa kosmoseagentuur. Euroopa kosmoseagentuuri arendus- ning tugikeskus rus. <https://rus-copernicus.eu/portal/>, 2017. Vaadatud: 2019-05-19.
- [67] Andriy Lazorenko. Tensorflow performance test: Cpu vs gpu. <https://medium.com/@andriylazorenko/tensorflow-performance-test-cpu-vs-gpu-79fcd39170c>, 2017. Vaadatud: 2019-05-19.
- [68] Eric Tsang AnujŠhah. Keras-el ja tensorflow-il põhinev 3d cnn implementatsioon. <https://github.com/Ectsang/3D-CNN-Keras>, 2016. Vaadatud: 2019-04-04.
- [69] Anuj Shah. 3d cnn youtube õpetus. [https://www.youtube.com/watch?v=dYVmeJFA\\_1A](https://www.youtube.com/watch?v=dYVmeJFA_1A), 2016. Vaadatud: 2019-04-04.
- [70] Harrison Jansma. Batch normalization-i mõju seletav õpetus. <https://towardsdatascience.com/dont-use-dropout-in-convolutional-networks-81486c823c16>, 2018. Vaadatud: 2019-04-04.
- [71] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15:1929–1958, 2014.
- [72] Scikit learn vabavaraline teek. Scikit-learn klassidele kaalude loomine, et tasakaalustada treeningprotsessi. [https://scikit-learn.org/stable/modules/generated/sklearn.utils.class\\_weight.compute\\_class\\_weight.html](https://scikit-learn.org/stable/modules/generated/sklearn.utils.class_weight.compute_class_weight.html), 2007. Vaadatud: 2019-05-19.

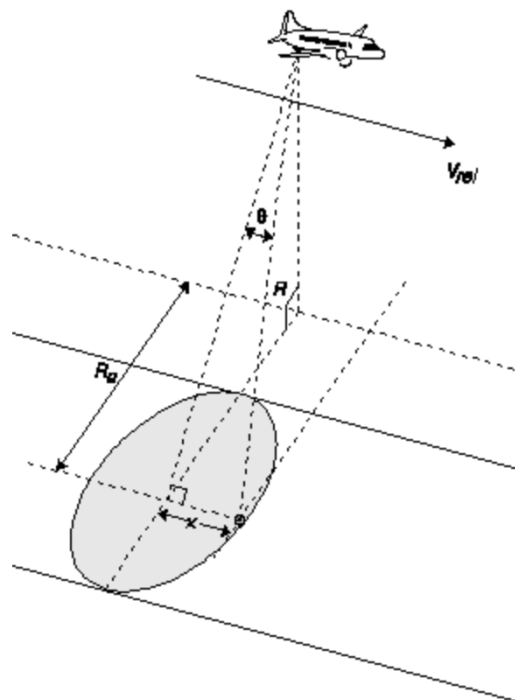
- [73] Diego Marcos, Michele Volpi, Nikos Komodakis, and Devis Tuia. Rotation equivariant vector field networks. *CoRR*, abs/1612.09346, 2016.
- [74] Carole H. Sudre, Wenqi Li, Tom Vercauteren, Sébastien Ourselin, and M. Jorge Cardoso. Generalised dice overlap as a deep learning loss function for highly unbalanced segmentations. *CoRR*, abs/1707.03237, 2017.
- [75] Sara Hooker, Dumitru Erhan, Pieter-Jan Kindermans, and Been Kim. Evaluating feature importance estimates. *CoRR*, abs/1806.10758, 2018.
- [76] Frank Paul. Tehisavaradari füüsikalised alused. [http://www.geo.uzh.ch/~fpaul/sar\\_theory.html](http://www.geo.uzh.ch/~fpaul/sar_theory.html), 1997. Vaadatud: 2019-05-26.

# Lisa A

## Lisad

### A.1 Tehisavaradari tööpõhimõte

Tehisavaradar (ingl. k. Synthetic-aperture radar ehk SAR) on radar, mis suudab teha teistest radaritest suurema lahutusvõimega pilte, kasutades ära radari antenni liikumist.



Joonis A.1: Tehisavaradari antenni kandev õhusõiduk liigub kiirusega  $v_{rel}$  asimuudi sihis. Asimuudi sihiga risti on ulatuse siht. [76]

Käsitleme kiiruseid õhusõiduki suhtes, ehk vaatleme õhusõidukit kui paigalolevat objekti. Kui

laineallikas liigub detektori poole, siis esineb Doppleri efekt:

$$f_o = \left[ \frac{1 + (v/c)}{1 - (v/c)} \right]^{1/2} f_s \quad (\text{A.1})$$

$f_0$  on paigalseisva objekti poolt kuulnud sagedus,  $f_s$  on laineallika tekitatud sagedus, ning  $v$  on nende objektide vaheline suhteline kiirus.

Kui kaks punkti maapeal on asimuudi sihis üksteisest kaugusel  $x$ , siis on nad vähesel määral erinevate nurkade all antenni suhtes, mistõttu ka nende suhtelised kiirused erinevad. Seega on nendest punktidest peegelduva raadiosignaali sagedused alati nihutatud doppleri efekti tõttu võrreldes antenni väljasaadetud signaaliga.

Leiame R-suunalise  $V_{rel}$  komponendi, ehk kiiruse, millega laineallikas läheneb antennile.  $V_{rel}$ -i saab projitseerida R-iga paralleelsele joonele, et saada:

$$v = v_{rel} \cos \left( \frac{\pi}{2} - \theta \right) = v_{rel} \sin \theta = v_{rel} \frac{x}{R} \quad (\text{A.2})$$

Lihtsustame  $v \ll c$  :

$$f_0 = \left( 1 + \frac{v}{c} \right) f_s \quad (\text{A.3})$$

Ning sageduse erinevus antenni ja punkti peegeldunud raadiolaine vahel on:

$$f_0 - f_s = \left( 1 + \frac{v}{c} \right) f_s - f_s = \frac{v}{c} f_s \quad (\text{A.4})$$

Tagasipeegelduse jaoks on Doppleri nihe samasugune, seega kokku on nihe:

$$f_d = \frac{2v}{c} f_s = \frac{2v_{rel} \sin \theta}{c} \cdot \frac{c}{\lambda} = \frac{2v_{rel} x}{\lambda R} \quad (\text{A.5})$$

Saadud kogunihe võimaldab leida täpselt, kust peegeldunud signaal pärineb, sest aeg, millega elektromagnetlaine levib kauguse  $2R$  on  $t = \frac{2R}{c}$ . Seega, kui koond doppleri sagedusnihe on  $f_{d1}$ , siis asimuudi koordinaat on:

$$x_1 = \frac{f_{d1} \lambda R}{2v_{rel}} \quad (\text{A.6})$$

Kuna õhusõiduki liikumine on ainult asimuudi suunaline, siis on võimalik saada suuremat lahutusvõimet ainult asimuudi sihis, ehk ulatus sihis jääb lahutusvõime algsega võrdseks.

Valemist (1.5) saab asimuudi lahutusvõime arvutada järgnevalt:

$$\delta x = \left( \frac{\lambda R}{2v_{rel}} \right) \delta f_d \quad (\text{A.7})$$

kus  $\delta f_d$  on ligikaudu aeg, mille uuritav punkt oli radari kimbu sees.  $\delta f_d \approx \frac{1}{t_\theta}$ . Seega leiame  $t_\theta$

$$t_\theta = \frac{\text{kaare pikkus}}{v_{rel}} = \frac{R\theta_H}{v_{rel}} = \frac{R, \lambda}{L_a v_{rel}} \quad (\text{A.8})$$

Kus  $L_a$  on antenni pikkus. Kokku saadakse lahutusvõimeks:

$$\delta x = \left( \frac{\lambda R}{2v_{rel}} \right) \left( \frac{L_2 v_{rel}}{R\lambda} \right) = \frac{L_a}{2} \quad (\text{A.9})$$

Saadud valem viitab, et mida väiksem on antenni pikkus, seda paremat ruumilist lahutusvõimet saadakse. Antenni pikkus on määratud tegelikult nõudega, et antenn suudaks signaale saata sagedusel  $\frac{\lambda}{L_a}$ . Selleks peab antenn olema piisavalt pikk, et antenni dipoolid moodustaks korrektse interferentsmusteri.

Tulemus  $\delta x = \frac{L_a}{2}$  pole päris täpne, sest esiteks juba eeldasime Doppleri efekti mitte-relatiivset kuju, kuid ka seepärast, et  $\delta f_d$  pole tegelikult konstante, vaid ajast sõltuv.

Tehisavaradarite mõõtmistel kasutatakse polariseeritud signaale, kuna paljude ainete vastasmõju tõttu polarisatsioon muutub ning seekaudu on võimalik saada lisainfot.

[76]

## A.2 Keskkonna ülesseadmine

Virtuaalse keskkonna ülesseadmiseks tuli käivitada järgnevad käsud antud järjekorras:

```
conda create -n aksel python=3.6.7 anaconda
conda install 'poppler<0.62'
conda install -c conda-forge 'georaster<1.25'
conda install gdal=2.3.2
conda install -c conda-forge notebook
conda install -c conda-forge keras
conda install -c menpo opencv=3
conda install -c conda-forge matplotlib
conda install pandas
conda install seaborn
conda install scikit-learn
conda install scikit-image
conda install pyshp
```

conda install shapely

conda install -c conda-forge rasterio

## A.3 Klassifitseeritavad klassid

Klassi #	Klassi nimetus	Pikslite arv
0	Mitte-põllukultuur	45843792
1	Aedmaasikas	7860
2	Astelpaju	5210
3	Heintaimed, kõrrelised	6689370
4	Heintaimed, liblikõielised	4940800
5	Kaer	719450
6	Kanep	113140
7	Karjatamine väljaspool põllumajanduslikku maad	32232
8	Kartul	75900
9	Mais	691568
10	Mustkesa	230890
11	Muu	210978
12	Põldhernes	1470090
13	Põlduba	432346
14	Peakapsas	530
15	Porgand	434
16	Punapeet	360
17	Rukis	402612
18	Sööti jäetud maa	51282
19	Suvinisu ja speltanisu	3448118
20	Suvioder	7085340
21	Suviraps ja -rüps	2175592
22	Suvitritikale	10408
23	Talinisu	4090562
24	Talioder	426114
25	Taliraps ja -rüps	1719380
26	Talitritikale	40886
27	Tatar	54484

## A.4 Github Kaustade struktuur

### **Juurkaust ./**

Sisaldab masinõppeks vajalikke python OOP klasse: ./s2\_preprocessor.py, ./s2\_model.py, ./data\_generator.py ning graafikute kuvamise klassi ./plotter.py.

Sisaldab fit() funktsiooni kasutavat masinõppe treenimise skripti ./fit\_main.py.

Sisaldab fit\_generator() funktsiooni kasutavat masinõppe treenimise skripti ./generator\_main.py.

Sisaldab ennustuste tegemise skripti ./predict.py.

Sisaldab klasside kaalude .npy faile.

Sisaldab funktsioone uurimisalasse kuuluvate töötletud ruutude listi tekitamiseks ./read\_processed\_tiles.py ning uurimisalasse kuuluvate ruutude objekte processed\_tile\_dict.npy ja zero\_tile\_dict.npy

### **S2\_products**

Siia tuleb panna Sentinel-2 pildid .SAFE formaadis.

### **Masks**

Siia tuleb panna klassifitseeritavate klasside põldude vektorkihid .shp formaadis.

### **ROI**

Siia tuleb panna uuritava huviala vektorkiht .shp formaadis. NB! kõikidel .shp failidel peab olema sama koordinaatsüsteem.

### **Gpt\_scripts**

Sisaldab bash skripte, mis kasutavad omakorda SNAP gpt-d, et töödelda sisendandmeid ning klasside maske.

### **Data**

Siia tekitab ./Scripts/bands.sh Sentinel-2 piltidest väljundfailid.

### **Label\_tifs**

Siia tekitab ./Scripts/label\_mask.sh klasside rasterfailid.

### **Predictions**

Sisaldab ./predict.py poolt tekitatud ennustuse tulemusi ning graafikuid.



Lisaks sisaldab ennustuste visualiseerimiseks vajalikke käsurea skripte, vea-maatriksi tekitamise skripti ja kogutäpsuse arvutamise skripti.

## Models

Sisaldab treenitud mudeleid ning treenimiste metaandmeid.

Lisaks sisaldab mudelite treenimise ajaloo ning metaandmete lugemise skripte

## Input\_data

Sisaldab ./Scripts/create\_input\_data.py poolt loodavaid 128x128 sisendtensorite ning tõeväärtuste binaarfaile.

## Big\_tile\_data

Sisaldab ./Scripts/create\_input\_data.py poolt loodavaid 512x512 sisendtensorite ning tõeväärtuste binaarfaile.

Sisaldab skripti kõige rohkemate erinevate klassidega ruudu leidmiseks ning skripti ruudul esinevate klasside loetlemiseks.

## Scripts

Sisaldab ./Input\_data/ ja ./Big\_tile\_data/ kausta sisendandmete tekitamise skripti ./Scripts/create\_input\_data.py.

Sisaldab skripte klasside kaalude arvutamiseks: count\_class\_occurrences.py, calculate\_weights.py ning vastavaid

Sisaldab aktivatsioonide visualiseerimise skripti.

## A.5 Töös kasutatud normeerimine

```
tile -= np.median(tile)
tile /= np.percentile(tile, 99)
np.putmask(tile, tile > 1, 1)
np.putmask(tile, tile < -1, -1)
```

Joonis A.2: Sisendandmete nullist üheni normeerimiseks kasutatud kood.

## A.6 Tõeväärtuste eeltöötlus SNAPi gpt käsureaga.



Joonis A.3: Siltide loomiseks kasutatud SNAP töötlusgraafid. Esmalt luuakse vasakpoolses graafis korrektne lahutusvõimega 5120x5120 ruut ning seejärel jooksutatakse parempoolset graafi iga klassi jaoks, et välja lõigata ning salvestada ainult ühte klassi kuuluvad pikslid.

```
#!/bin/bash
echo "===== Phase 1 ====="
#Generate .dim files, which are needed to make label masks

GPT_PATH=/sar/aksel/snap/snap/bin/gpt
OUTPUT_PATH=/sar/aksel/Bachelor_thesis/Gpt_scripts/
INPUT_PATH=/sar/aksel/Bachelor_thesis/S2_Products/

#NB! The generated files will get the suffix '_y-x' where y will show the tile's number along y-axis and x along x-axis. This is so that tile location notation would be the same as the way python 2D arrays are indexed.
for product in $(ls $INPUT_PATH); do
    index=$((#product)-4))
    product_file_extension="${product:$index:4}"
    prefix_product=${product:11:-46}
    #break #Uncomment this line to skip Phase 1
    if [ "${product_file_extension}" = "SAFE" ]; then
        for (( k = 0; k < 2; k++ )); do
            for (( l = 0; l < 2; l++ )); do
                y=$(expr 5120 '*' "$l")
                x=$(expr 5120 '*' "$k" + 740 )
                subset_parameters="$x,$y,5120,5120"
                sed "s@{input_safe}@${INPUT_PATH}${product}@g; s@{subset_params}@${subset_parameters}@g; s@{p
refix_dim}@${OUTPUT_PATH}${prefix_product}_${l}-${k}@g" label_mask_preprocessing.in > label_mask_preprocessing_out.xml
                $GPT_PATH label_mask_preprocessing_out.xml -c 4G -q 3 -x -J-Xmx15G -J-Xms4G
            done
        done
    fi
done

echo "===== Phase 2 ====="
#Make label mask tifs.
#This script throws an error, when there are no fields in the current tile. That means no file will be made for that class/tile combination. We will generate empty files in Phase 3

MASK_PATH=/sar/aksel/Bachelor_thesis/Masks/
OUTPUT_PATH=/sar/aksel/Bachelor_thesis/Label_tifs/
for (( k = 0; k < 2; k++ )); do
    for (( l = 0; l < 2; l++ )); do
        #break #Uncomment this line to skip Phase 2
        INPUT_PATH=/sar/aksel/Bachelor_thesis/Gpt_scripts/${prefix_product}_${l}-${k}.dim
        for mask_file in $(ls $MASK_PATH); do
            index=$((#mask_file)-3))
            mask_file_extension="${mask_file:$index:3}"
            if [ "${mask_file_extension}" = "shp" ]; then
                mask=${mask_file:1:-4}
                echo $mask
                prefix_mask="Mask_${mask}"
                sed "s@{input_dim}@${INPUT_PATH}@g; s@{mask_file}@${MASK_PATH}${mask_file}@g; s@{mask}@${mask
k}@g; s@{prefix_mask}@${OUTPUT_PATH}${prefix_mask}_${l}-${k}@g" label_mask_preprocessing_2.in > label_mask_preprocessing_out_2.xml
                $GPT_PATH label_mask_preprocessing_out_2.xml -c 4G -q 3 -x -J-Xmx15G -J-Xms4G 2>>error_tiles.txt
                echo "${mask}_${l}-${k}">>error_tiles.txt
            fi
        done
    done
done

echo "===== Phase 3 ====="
#Generate empty files, so that during training, the amount of classes would still be the same for every tile.
for missing_file in `sed -n '/Error/{n;p}' error_tiles.txt`; do
    touch ${OUTPUT_PATH}Mask_${missing_file}.tif
done
```

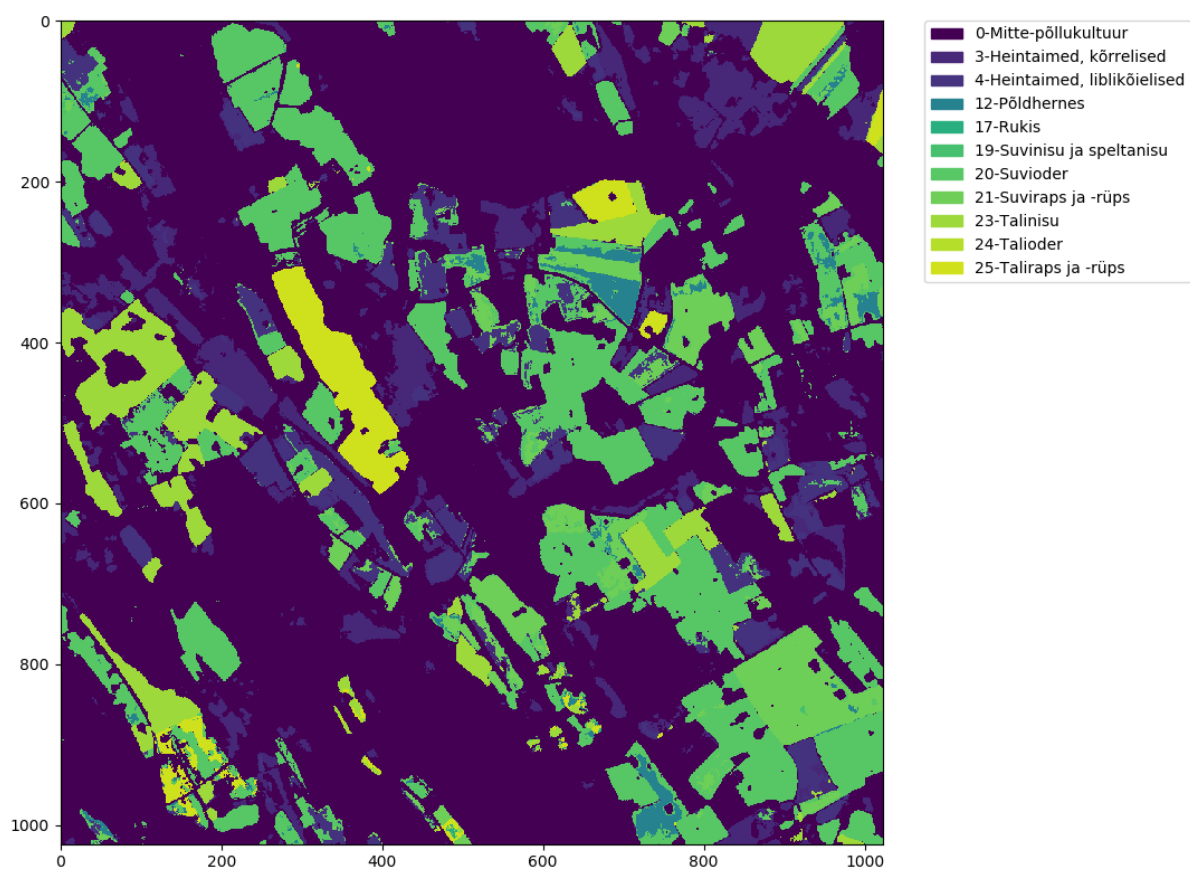
Joonis A.4: Tõeväärtuste ehk siltide tekitamiseks kasutatud skript. Vastab github hoidla failile `./Gpt_scripts/label_mask.sh`

## A.7 Normeerimata veamaatriks

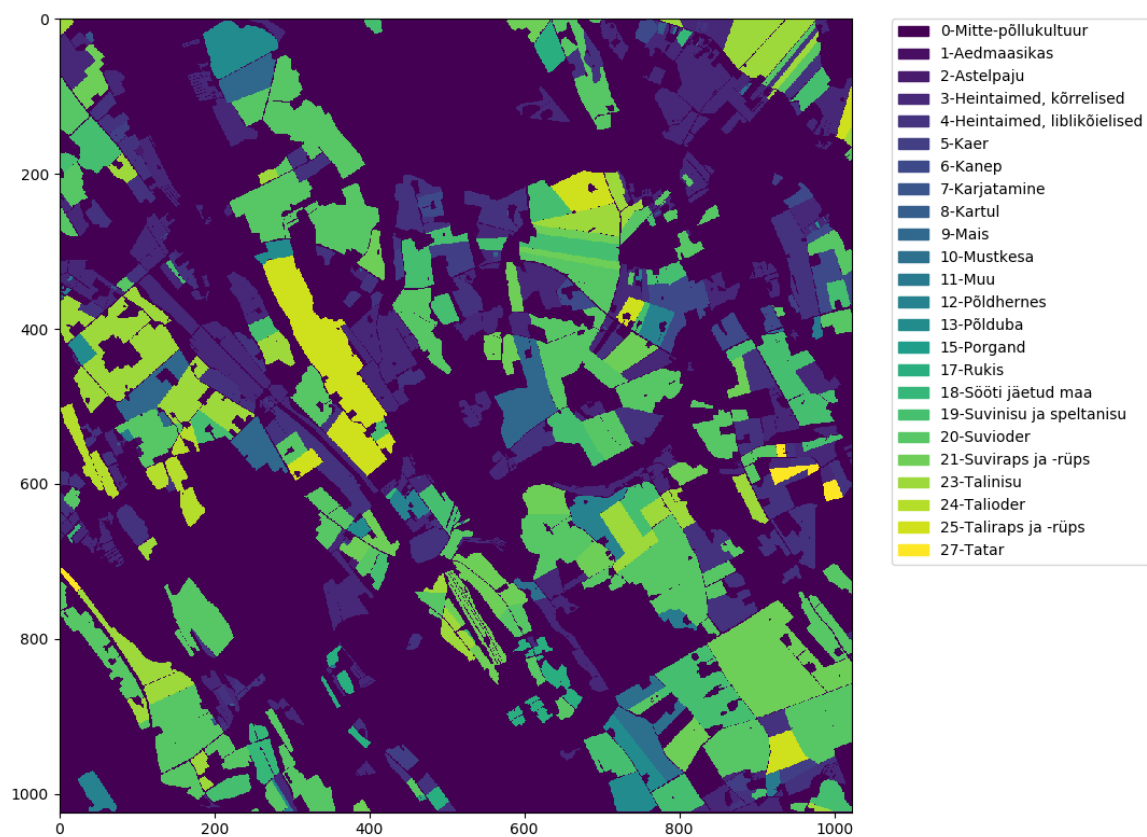
Tegelik klass	0-Mitte-põllukultuur	64646	22222	4670	472	25	9	7423	1281	10454	0	2385
	3-Heintaimed, kõrrelised	46617	85025	5975	1002	0	5	2590	909	568	0	29
	4-Heintaimed, liblikõielised	15593	14554	39730	719	1	6	6958	4650	881	0	56
	12-Põldhernes	2383	257	1303	9165	0	59	6902	3355	83	0	1
	17-Rukis	2922	829	787	200	1880	0	706	43	7082	1	1240
	19-Suvinisu ja speltanisu	5837	341	4292	2808	0	330	74101	12874	288	0	42
	20-Suvioder	11634	218	7626	4920	2	68	184221	6724	872	0	3493
	21-Suviraps ja -rüps	3609	501	4603	1989	0	74	13130	55009	80	0	0
	23-Talinisu	6016	477	2602	170	16	0	7605	245	122582	0	4433
	24-Talioder	450	136	175	0	0	0	3	0	11590	3	29
	25-Taliraps ja -rüps	1594	968	1571	288	0	0	257	2061	1089	1	29529
	Ennustatud klass											

Joonis A.5: Parima närvivõrgu veamaatriks kümne 512x512 ruudu sisendandmetega.

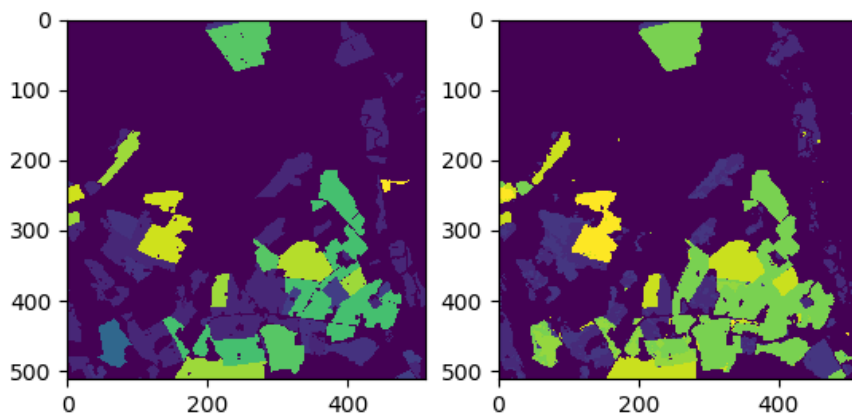
## A.8 Lõpliku mudeli ennustused



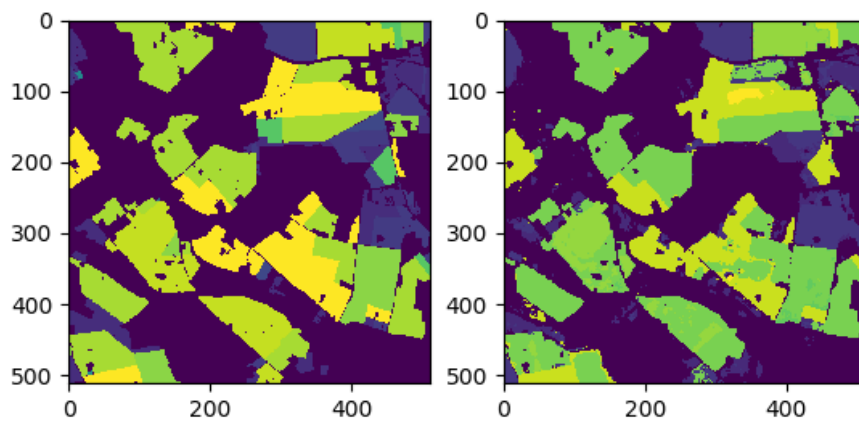
Joonis A.6: Parima närvivõrgu ennustus kõige tihedama põldude levialaga 1024x1024 ruudul. Kogutäpsus 75,87%.



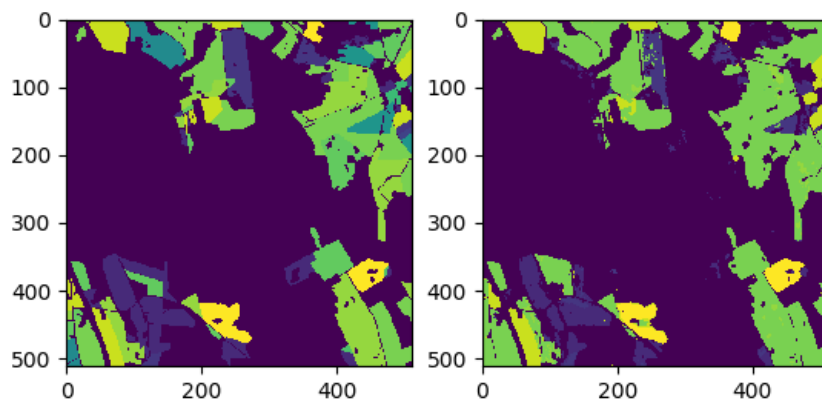
Joonis A.7: Kõige tihedama põldude levialaga 1024x1024 ruudu tõeväärtused.



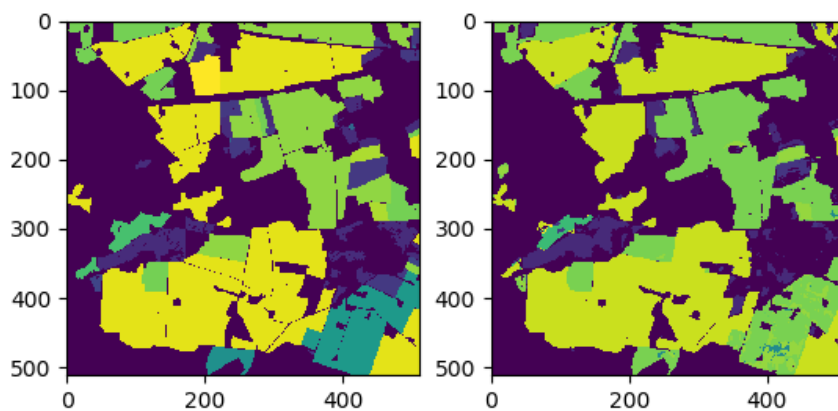
Joonis A.8: Kogutäpsus 80,89%, 0-klassi protsent 69,72%.



Joonis A.9: Kogutäpsus 80,59%, 0-klassi protsent 49,10%.



Joonis A.10: Kogutäpsus 81,18%, 0-klassi protsent 66,50%.



Joonis A.11: Kogutäpsus 82,52%, 0-klassi protsent 43,96%.

# **Lihtlitsents lõputöö reprodutseerimiseks ja lõputöö üldsusele kättesaadavaks tegemiseks**

Mina, Aksel Allas,

1. annan Tartu Ülikoolile tasuta loa (lihtlitsentsi) enda loodud teose

## **Satelliidipiltidelt põllukultuuride pikselhaaval klassifitseerimine 3D sidumnärvivõrguga,**

mille juhendajad on PhD Kaupo Voormansik ja MSc Indrek Sünter, reprodutseerimiseks eesmärgiga seda säilitada, sealhulgas lisada digitaalarhiivi DSpace kuni autoriõiguse kehtivuse lõppemiseni.

2. Annan Tartu Ülikoolile loa teha punktis 1 nimetatud teos üldsusele kättesaadavaks Tartu Ülikooli veebikeskkonna, sealhulgas digitaalarhiivi DSpace kaudu Creative Commons'i litsentsiga CC BY NC ND 3.0, mis lubab autorile viidates teost reprodutseerida, levitada ja üldsusele suunata ning keelab luua tuletatud teost ja kasutada teost ärieesmärgil, kuni autoriõiguse kehtivuse lõppemiseni.
3. olen teadlik, et punktis 1 ja 2 nimetatud õigused jäävad alles ka autorile.
4. kinnitan, et lihtlitsentsi andmisega ei rikuta teiste isikute intellektuaalomandi ega isikuandmete kaitse seadusest tulenevaid õigusi.

Aksel Allas,

Tartu, 31. mai 2019. a.